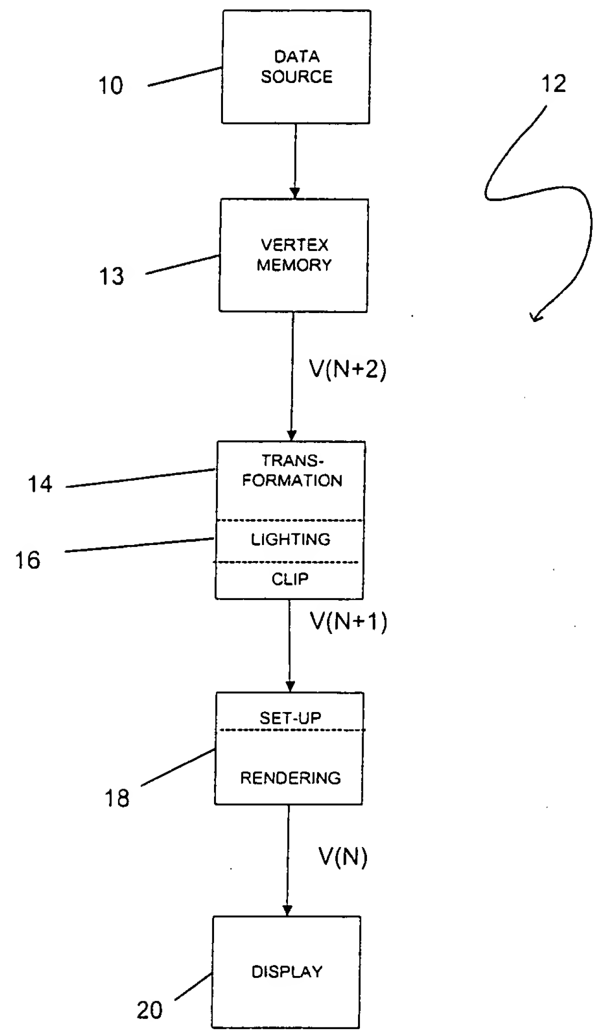
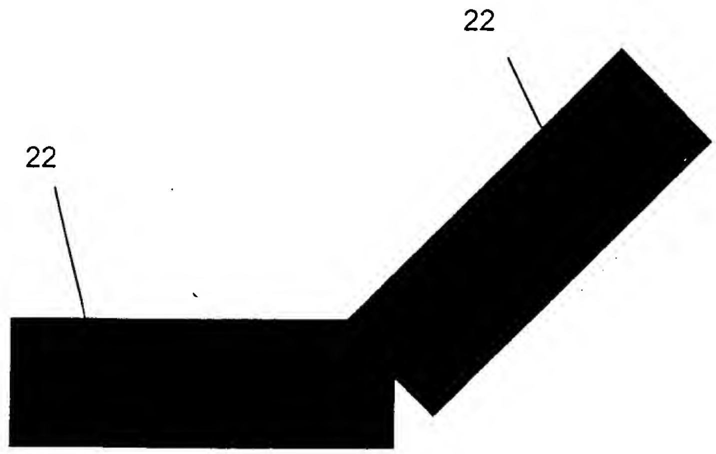
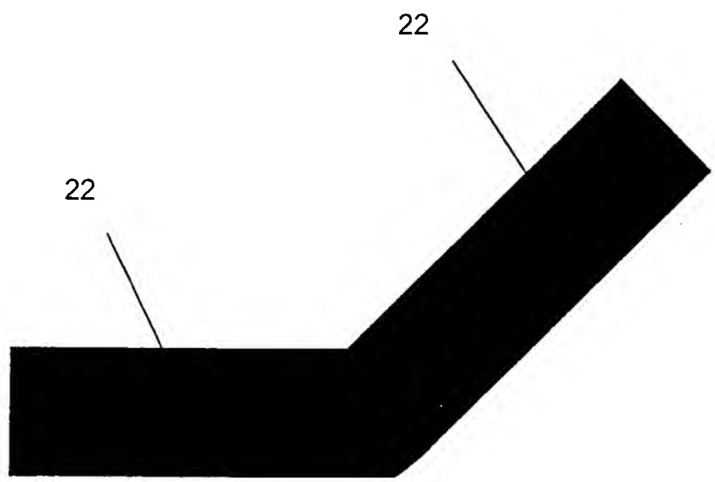


Figure 1
(PRIOR ART)



Before



After

Figure 1A

(PRIOR ART)

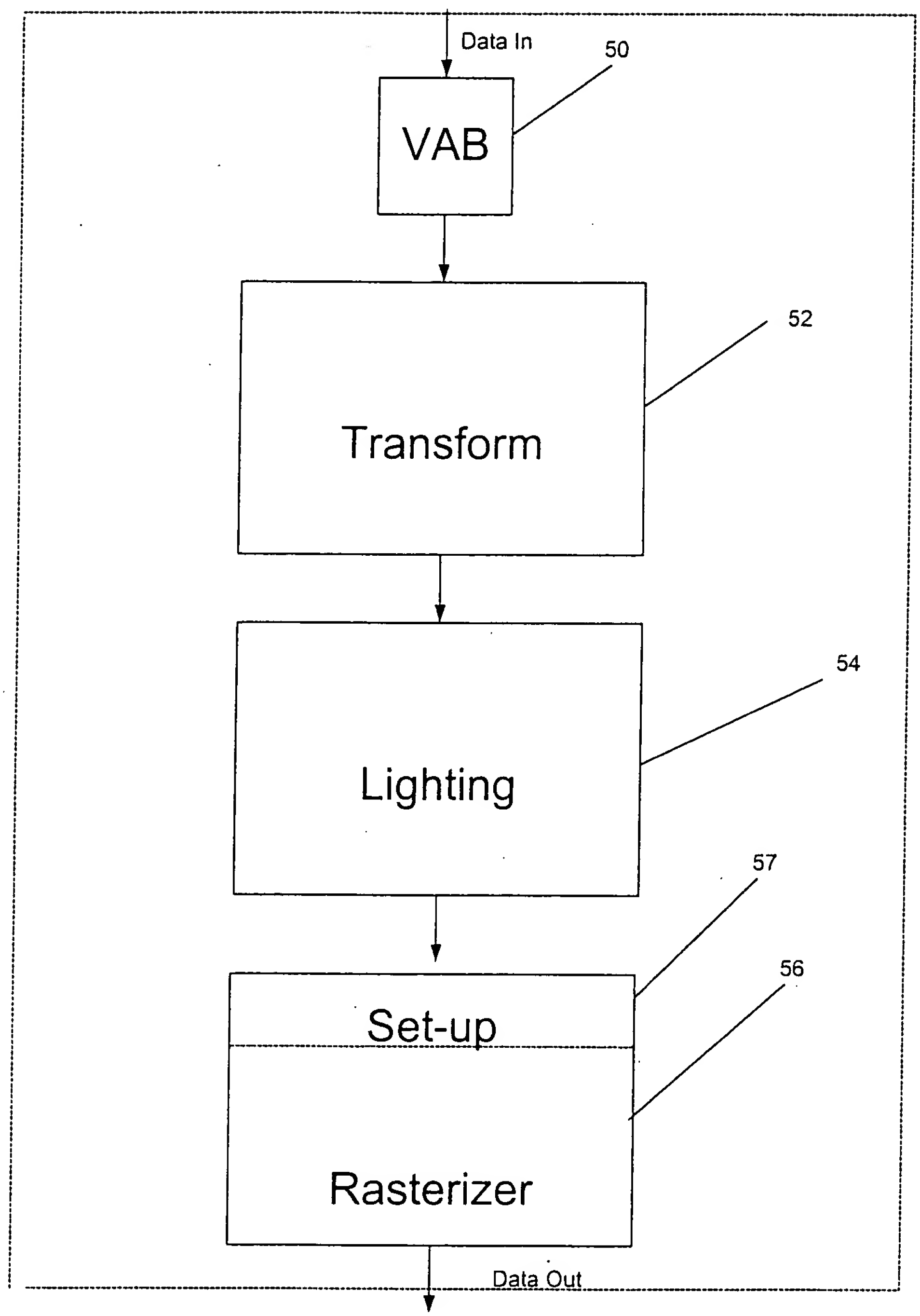

Data In
50
VAB
52
Transform
54
Lighting
57
56
Set-up
Rasterizer
Data Out

Figure 1B

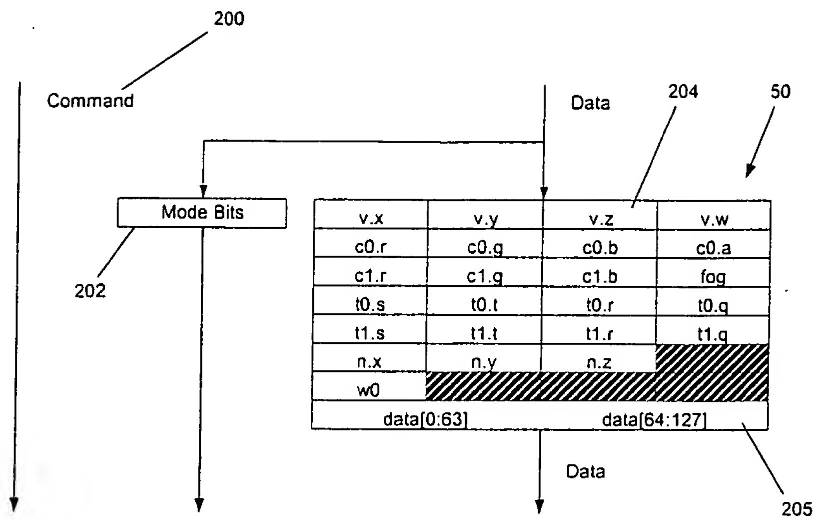


Figure 2

Command	Transform Stall	Lighting Stall	Description
FE2XF_CMD_NOP			No operation. Can be used as a spacer between commands
FE2XF_CMD_VERTEX	read	read	Vertex data.
FE2XF_CMD_PASSTHR			Passthrough. Transform and lighting pass the data through.
FE2XF_CMD_RDVAB			Read the VAB contents when context switching.
FE2XF_CMD_LDMODE			Load new mode bits.
FE2XF_CMD_LDXFCTX	write		Load transform context memory data
FE2XF_CMD_RDXFCTX	read		Read transform context memory data.
FE2XF_CMD_LDLTCTX		write	Load lighting context memory data.
FE2XF_CMD_RDLTCTX		read	Read lighting context memory data.
FE2XF_CMD_LDLTC0		write	Load lighting context0 memory data.
FE2XF_CMD_RDLTC0		read	Read lighting context0 memory data.
FE2XF_CMD_LDLTC1		write	Load lighting context1 memory data.
FE2XF_CMD_RDLTC1		read	Read lighting context1 memory data.
FE2XF_CMD_LDLTC2		write	Load lighting context2 memory data.
FE2XF_CMD_RDLTC2		read	Read lighting context2 memory data.
FE2XF_CMD_LTLTC3		write	Load lighting context3 memory data.
FE2XF_CMD_RDLTC3		read	Read lighting context3 memory data.
FE2XF_CMD_SYNC	read+write	read+write	Similar to NOP, but is not allowed to be processed in parallel.

Figure 2A

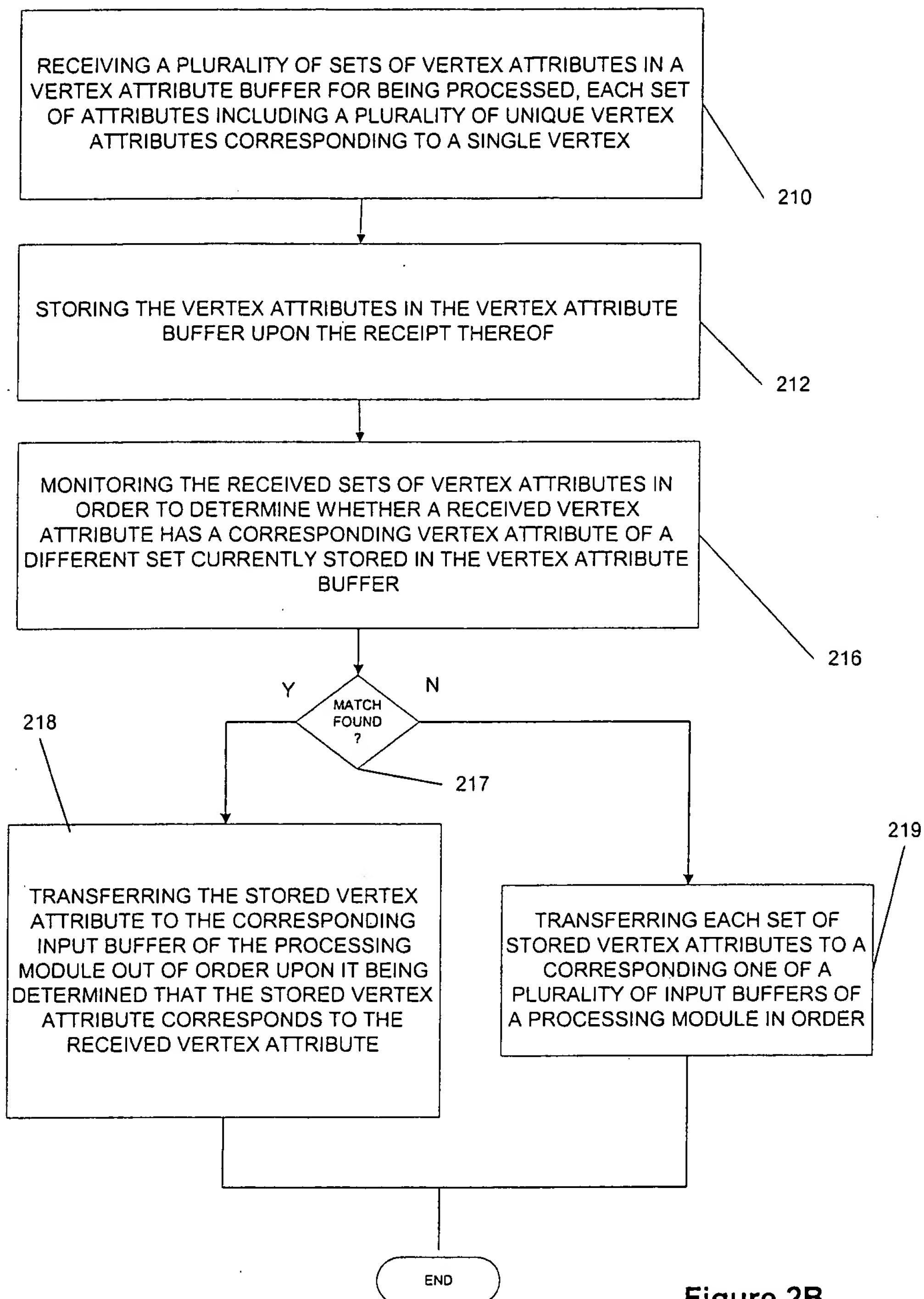

RECEIVING A PLURALITY OF SETS OF VERTEX ATTRIBUTES IN A VERTEX ATTRIBUTE BUFFER FOR BEING PROCESSED, EACH SET OF ATTRIBUTES INCLUDING A PLURALITY OF UNIQUE VERTEX ATTRIBUTES CORRESPONDING TO A SINGLE VERTEX
210
STORING THE VERTEX ATTRIBUTES IN THE VERTEX ATTRIBUTE BUFFER UPON THE RECEIPT THEREOF
212
MONITORING THE RECEIVED SETS OF VERTEX ATTRIBUTES IN ORDER TO DETERMINE WHETHER A RECEIVED VERTEX ATTRIBUTE HAS A CORRESPONDING VERTEX ATTRIBUTE OF A DIFFERENT SET CURRENTLY STORED IN THE VERTEX ATTRIBUTE BUFFER
216
Y
N
MATCH FOUND ?
217
218
TRANSFERRING THE STORED VERTEX ATTRIBUTE TO THE CORRESPONDING INPUT BUFFER OF THE PROCESSING MODULE OUT OF ORDER UPON IT BEING DETERMINED THAT THE STORED VERTEX ATTRIBUTE CORRESPONDS TO THE RECEIVED VERTEX ATTRIBUTE
219
TRANSFERRING EACH SET OF STORED VERTEX ATTRIBUTES TO A CORRESPONDING ONE OF A PLURALITY OF INPUT BUFFERS OF A PROCESSING MODULE IN ORDER
END

Figure 2B

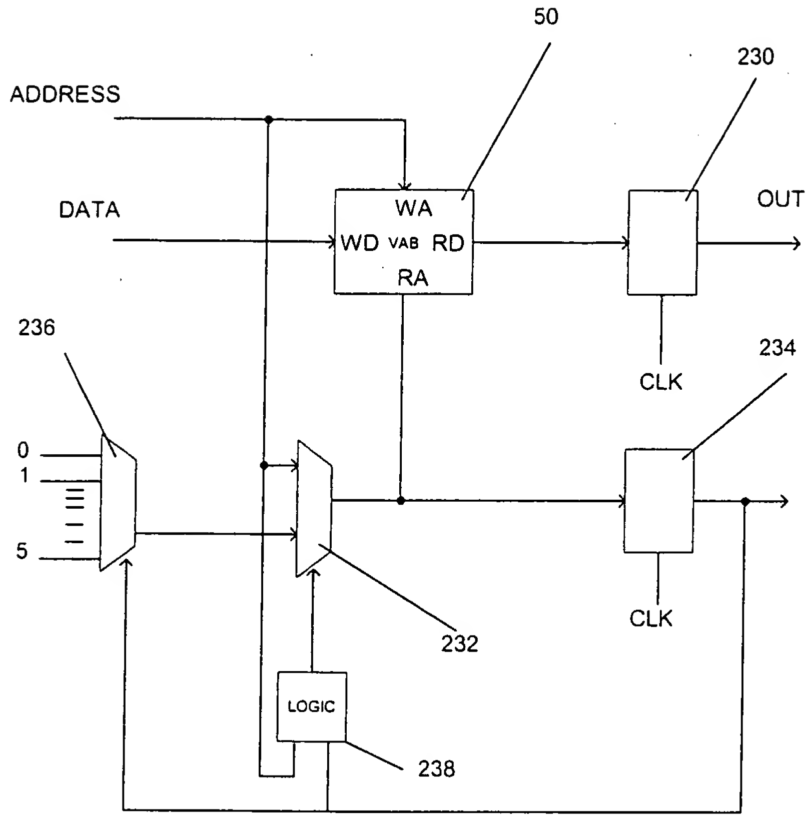
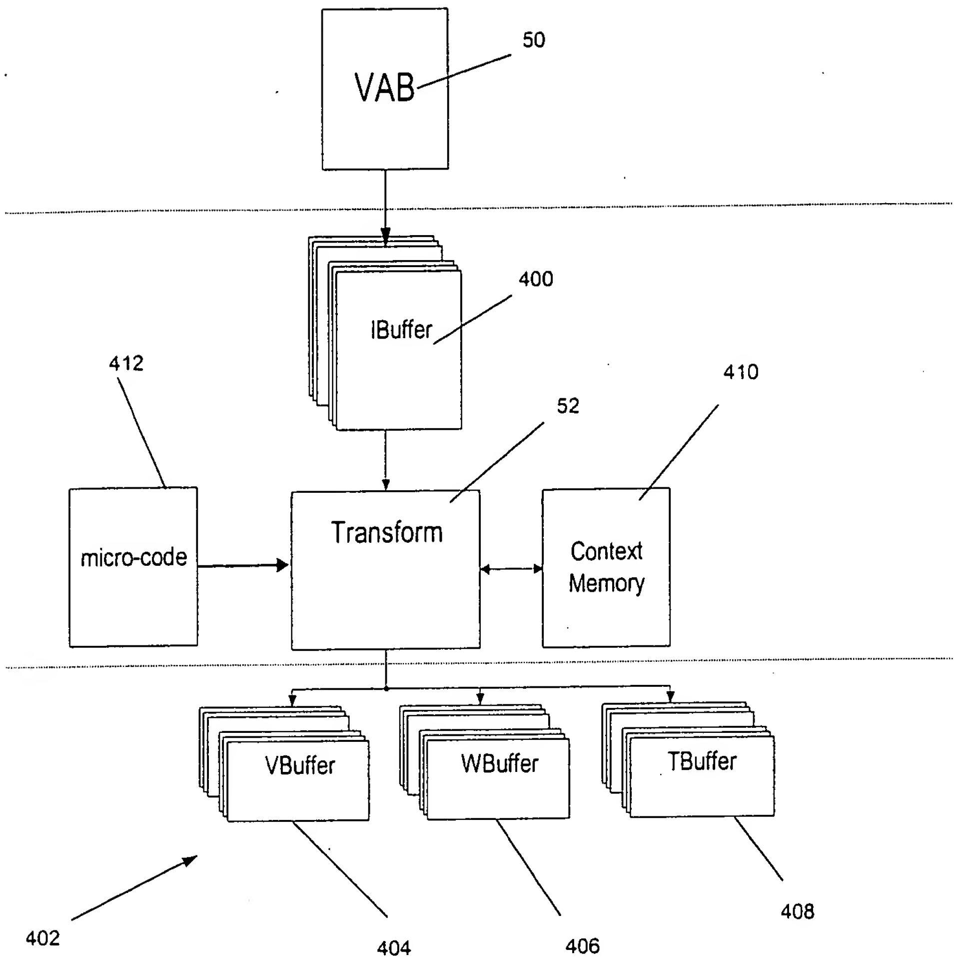

50
230
ADDRESS
DATA
WA
WD
VAB
RD
RA
OUT
236
CLK
234
0
1
5
232
CLK
LOGIC
238

Figure 2C

Mode Bit	Bits	Description
T0	1	Texture 0 enable
TXF0	1	Texture 0 matrix transform enable
TDV0	1	Texture 0 w divide enable
T0S	3	Texture 0 texgen s control
T0T	3	Texture 0 texgen t control
T0U	3	Texture 0 texgen r control
T0Q	2	Texture 0 texgen q control
T1	1	Texture 1 enable
TXF1	1	Texture 1 matrix transform enable
TDV1	1	Texture 1 w divide enable
T1S	3	Texture 1 texgen s control
T1T	3	Texture 1 texgen t control
T1U	3	Texture 1 texgen r control
T1Q	2	Texture 1 texgen q control
ETY	1	Eye type infinite(0) or local(1)
LIT	1	Lighting enable
NRM	1	Normal normalize enable
FOG	1	Fog enable
LIS	16	Light status (8 lights by 2 bits each, 0:off, 1:infinite, 2:local, 3: spotlight)
FG	2	Foggen control (0: off, 1: radial, 2: plane)
LAT	1	Light attenuation control (0: invert, 1: no invert)
C1I	1	Specular color input enable
C1O	1	Specular color output enable
CM	4	Color material control (1: emissive, 2:ambient, 4: diffuse, 8: specular)
PP	1	Point parameter enable
SKIN	1	Skinning enable
VPAS	1	Vertex pass enable

Figure 3



VAB
50
IBuffer
400
412
410
52
micro-code
Transform
Context Memory
VBuffer
WBuffer
TBuffer
402
404
406
408

Figure 4

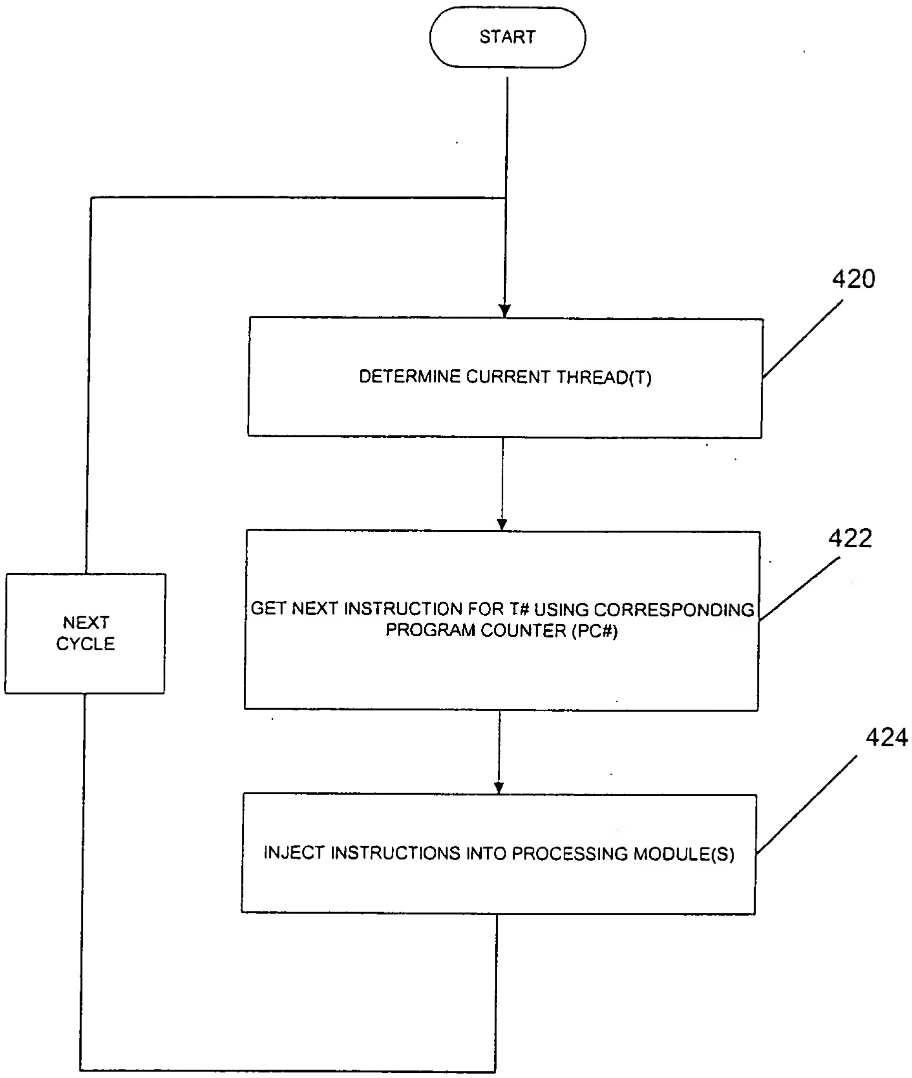

START
DETERMINE CURRENT THREAD(T)
420
GET NEXT INSTRUCTION FOR T# USING CORRESPONDING PROGRAM COUNTER (PC#)
422
INJECT INSTRUCTIONS INTO PROCESSING MODULE(S)
424
NEXT CYCLE

Figure 4A

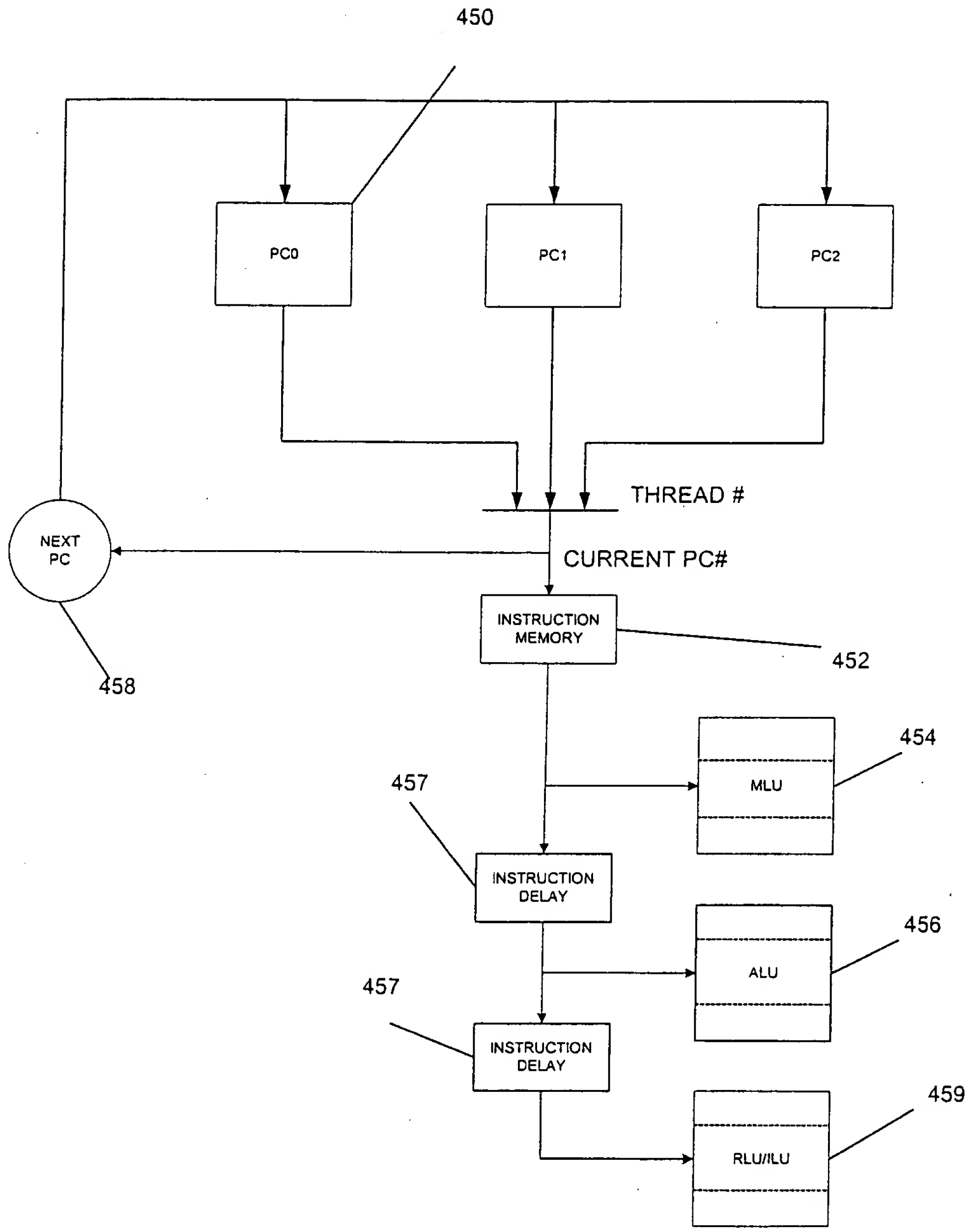

450
PC0
PC1
PC2
THREAD #
NEXT PC
CURRENT PC#
INSTRUCTION MEMORY
452
458
454
MLU
457
INSTRUCTION DELAY
456
ALU
457
INSTRUCTION DELAY
459
RLU/ILU

Figure 4B

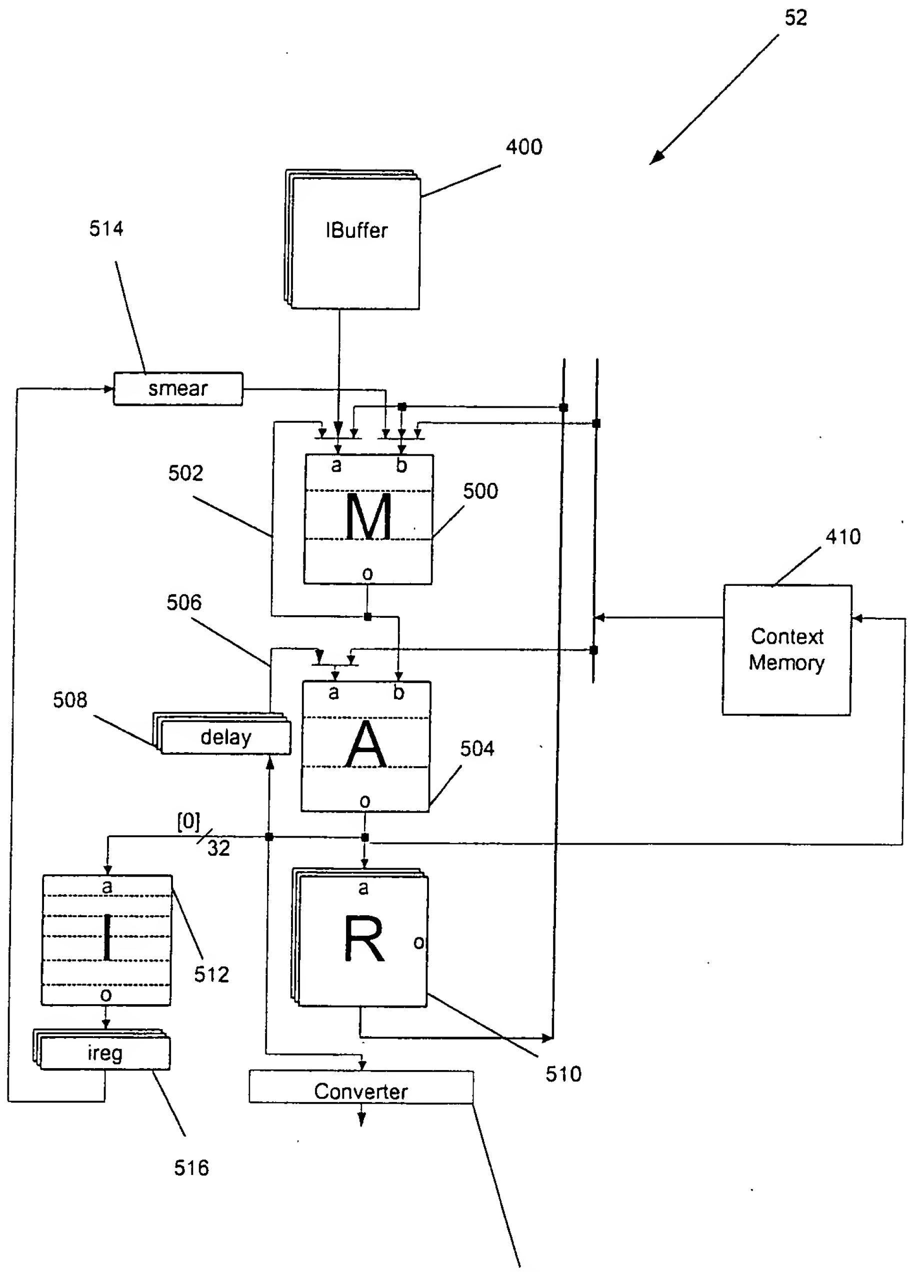

52
400
IBuffer
514
smear
a
b
M
o
502
500
410
Context
Memory
506
a
b
A
o
504
508
delay
[0]
32
a
I
o
512
a
R
o
ireg
516
Converter
510
518

Figure 5

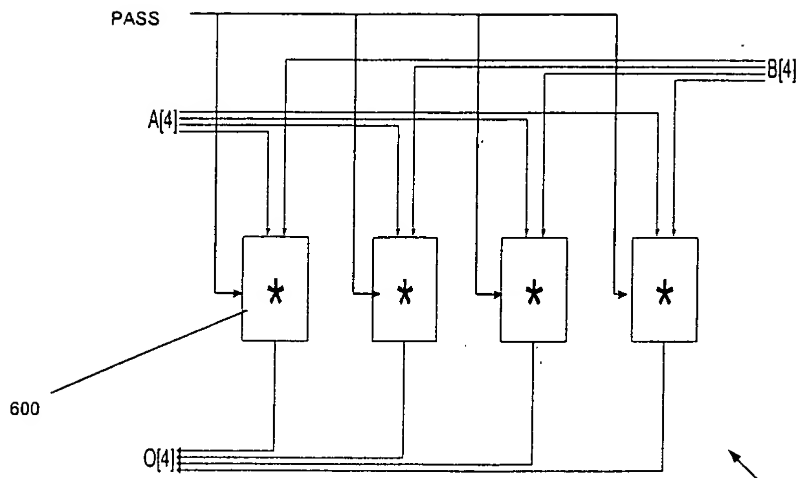


Figure 6

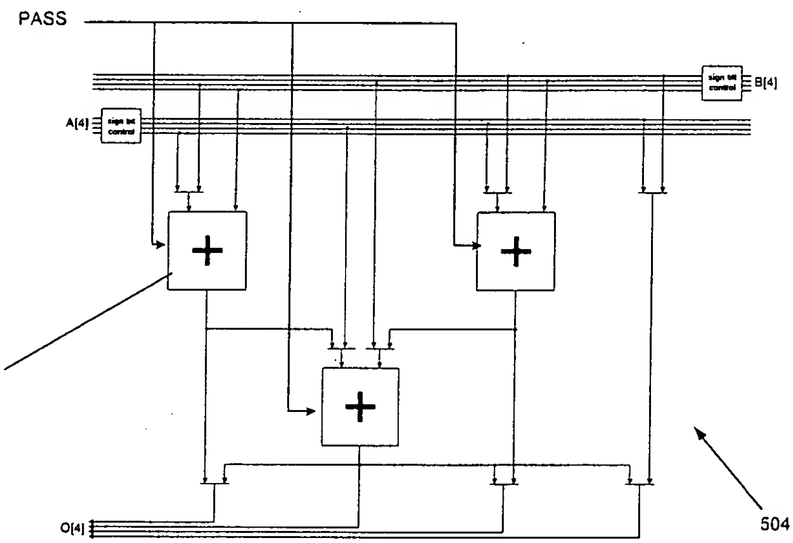


Figure 7

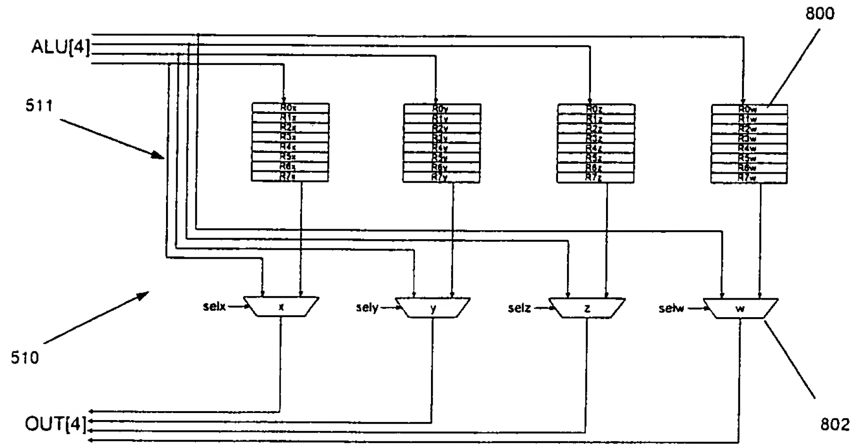


Figure 8

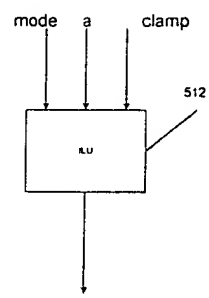


Figure 9

Address	Target	Action	Description
TPOS	TBUFFER	T[0] = ALU	Position
TT0	TBUFFER	T[3] = ALU	Texture0
TT1	TBUFFER	T[4] = ALU	Texture1
WEV	WBUFFER,VBUFFER	W[0] = ALU, V[0].y = ALU.w	Eye vector
WLV0	WBUFFER,VBUFFER	W[1] = ALU, V[1].y = ALU.w	Light0 direction vector
WLV1	WBUFFER,VBUFFER	W[2] = ALU, V[2].y = ALU.w	Light1 direction vector
WLV2	WBUFFER,VBUFFER	W[3] = ALU, V[3].y = ALU.w	Light2 direction vector
WLV3	WBUFFER,VBUFFER	W[4] = ALU, V[4].y = ALU.w	Light3 direction vector
WLV4	WBUFFER,VBUFFER	W[5] = ALU, V[5].y = ALU.w	Light4 direction vector
WLV5	WBUFFER,VBUFFER	W[6] = ALU, V[6].y = ALU.w	Light5 direction vector
WLV6	WBUFFER,VBUFFER	W[7] = ALU, V[7].y = ALU.w	Light6 direction vector
WLV7	WBUFFER,VBUFFER	W[8] = ALU, V[8].y = ALU.w	Light7 dirction vector
WSL0	WBUFFER	W[9] = ALU	Spotlight0 cos
WSL1	WBUFFER	W[10] = ALU	Spotlight1 cos
WSL2	WBUFFER	W[11] = ALU	Spotlight2 cos
WSL3	WBUFFER	W[12] = ALU	Spotlight3 cos
WSL4	WBUFFER	W[13] = ALU	Spotlight4 cos
WSL5	WBUFFER	W[14] = ALU	Spotlight5 cos
WSL6	WBUFFER	W[15] = ALU	Spotlight6 cos
WSL7	WBUFFER	W[16] = ALU	Spotlight7 cos
VED	VBUFFER	V[0].x = 1.0, V[0].z = ALU.w	Eye radial distance vector
VLD0	VBUFFER	V[1].x = 1.0, V[1].z = ALU.w	Light0 distance vector
VLD1	VBUFFER	V[2].x = 1.0, V[2].z = ALU.w	Light1 distance vector
VLD2	VBUFFER	V[3].x = 1.0, V[3].z = ALU.w	Light2 distance vector
VLD3	VBUFFER	V[4].x = 1.0, V[4].z = ALU.w	Light3 distance vector
VLD4	VBUFFER	V[5].x = 1.0, V[5].z = ALU.w	Light4 distance vector
VLD5	VBUFFER	V[6].x = 1.0, V[6].z = ALU.w	Light5 distance vector
VLD6	VBUFFER	V[7].x = 1.0, V[7].z = ALU.w	Light6 distance vector
VLD7	VBUFFER	V[8].x = 1.0, V[8].z = ALU.w	Light7 distance vector
VC0	VBUFFER,TBUFFER	V[9] = ALU, T[1] = ALU	Diffuse color
VC1	VBUFFER,TBUFFER	V[10] = ALU, T[2] = ALU	Specular color
VNRM	VBUFFER	V[11] = ALU	Normal vector
VED2	VBUFFER	V[12] = ALU	Eye planar distance vector
TVW_NOP			No valid output.

Figure 10

Microcode Field	Bits	Location	Delay	Description
oa	6	0: 5	2	Output buffer write address
rra	3	6: 8	0	RLU read address
rwm	4	9:12	2	RLU write mask
rwa	3	13:15	2	RLU write address
ilu	2	16:17	2	ILU operation
alu	4	18:21	1	ALU operation
ais	2	22:23	1	ALU sign control
aia	1	24	1	ALU input A mux
mlu	3	25:27	0	MLU operation
mib	2	28:29	0	MLU input B mux
mia	2	30:31	0	MLU input A mux
va	3	32:34	0	Input buffer read address
ce	1	35	0,2	Context memory read/write
ca	6	36:41	0,2	Context memory address
mr	2	42:43	0	MLU input vector rotate

Figure 11

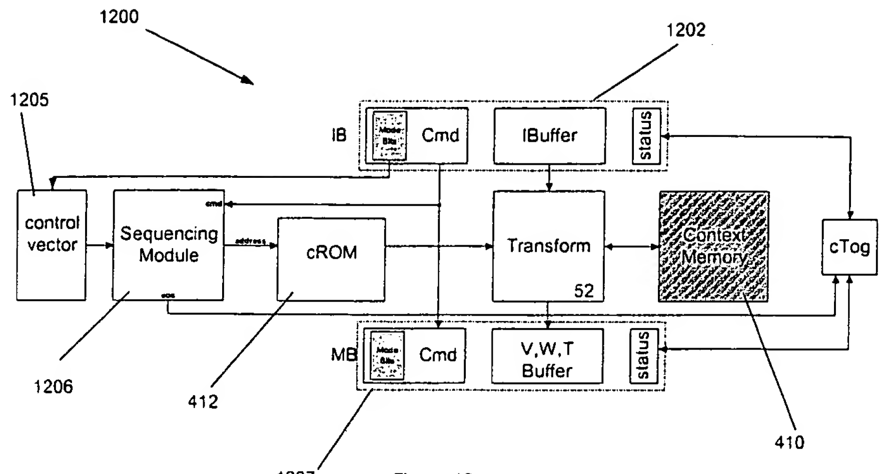

1200
1202
1205
IB
Cmd
IBuffer
status
control vector
Sequencing Module
cmd
address
cROM
Transform
52
Context Memory
cTog
eos
MB
Cmd
V,W,T Buffer
status
1206
412
410
1207

Figure 12

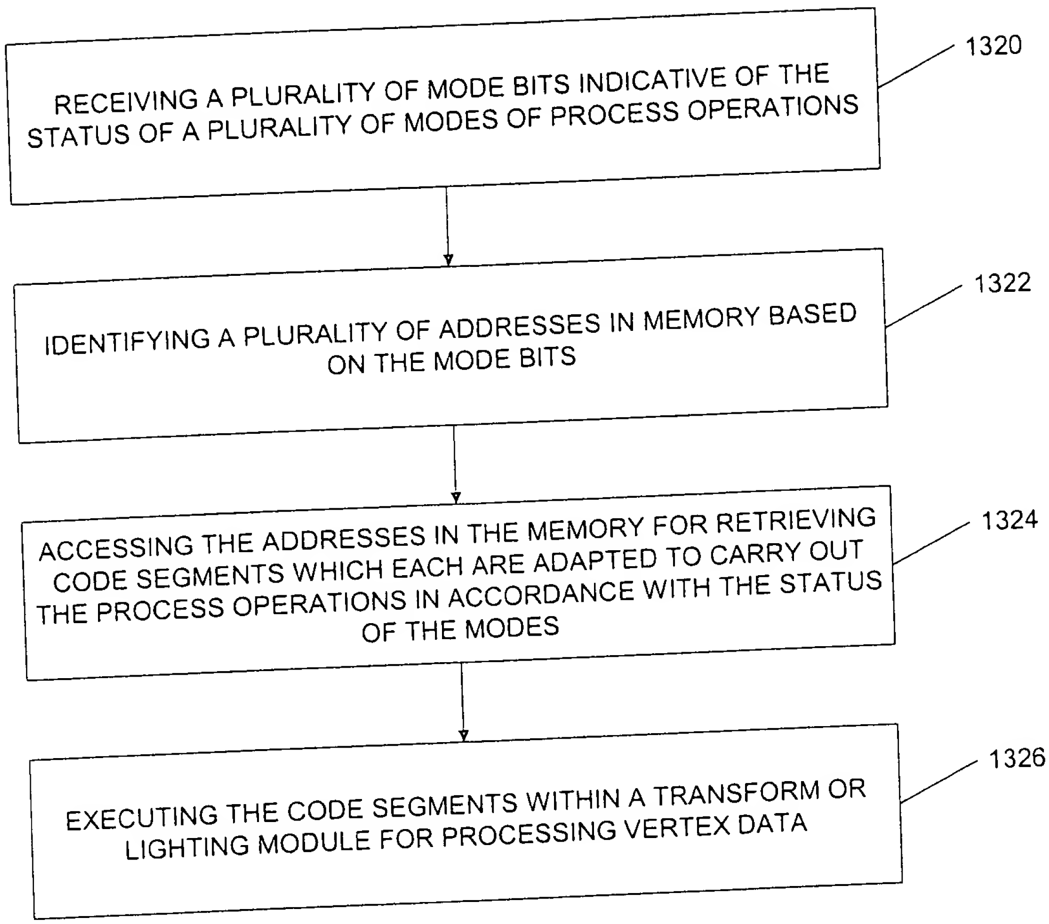

RECEIVING A PLURALITY OF MODE BITS INDICATIVE OF THE STATUS OF A PLURALITY OF MODES OF PROCESS OPERATIONS
1320
IDENTIFYING A PLURALITY OF ADDRESSES IN MEMORY BASED ON THE MODE BITS
1322
ACCESSING THE ADDRESSES IN THE MEMORY FOR RETRIEVING CODE SEGMENTS WHICH EACH ARE ADAPTED TO CARRY OUT THE PROCESS OPERATIONS IN ACCORDANCE WITH THE STATUS OF THE MODES
1324
EXECUTING THE CODE SEGMENTS WITHIN A TRANSFORM OR LIGHTING MODULE FOR PROCESSING VERTEX DATA
1326

Figure 13

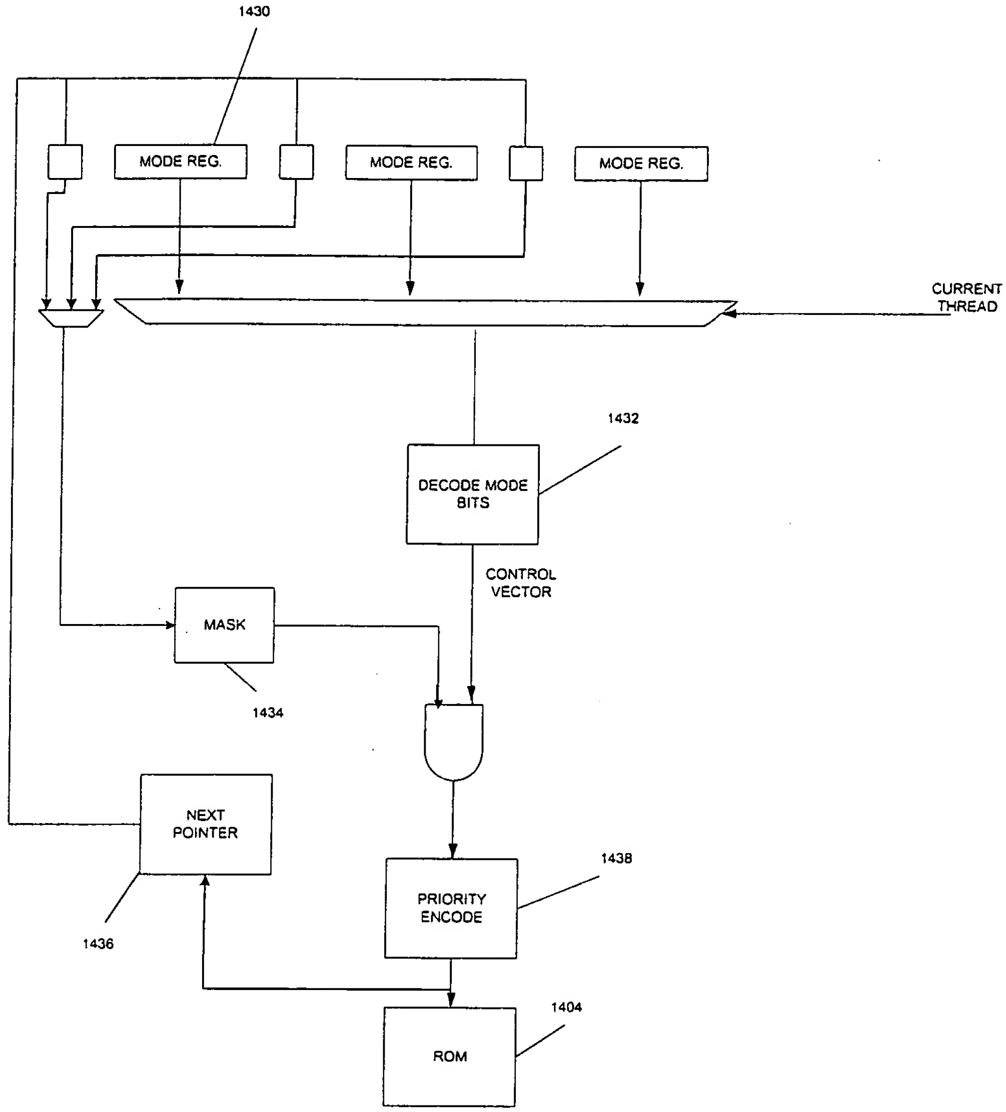

1430
MODE REG.
MODE REG.
MODE REG.
CURRENT THREAD
1432
DECODE MODE BITS
CONTROL VECTOR
MASK
1434
NEXT POINTER
1436
PRIORITY ENCODE
1438
ROM
1404

Figure 14

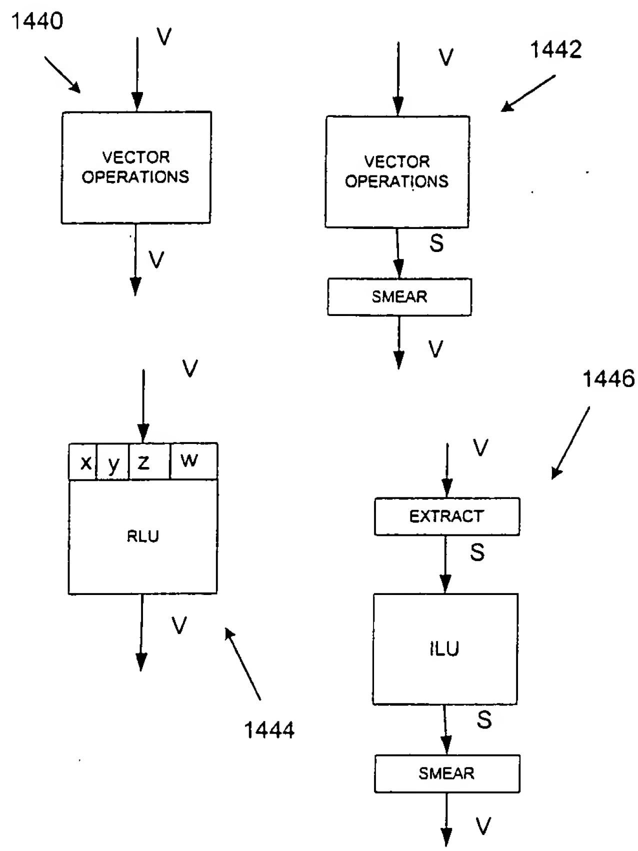

1440
V
VECTOR
OPERATIONS
V
1442
V
VECTOR
OPERATIONS
S
SMEAR
V
V
x
y
z
w
RLU
V
1444
1446
V
EXTRACT
S
ILU
S
SMEAR
V

Figure 14A

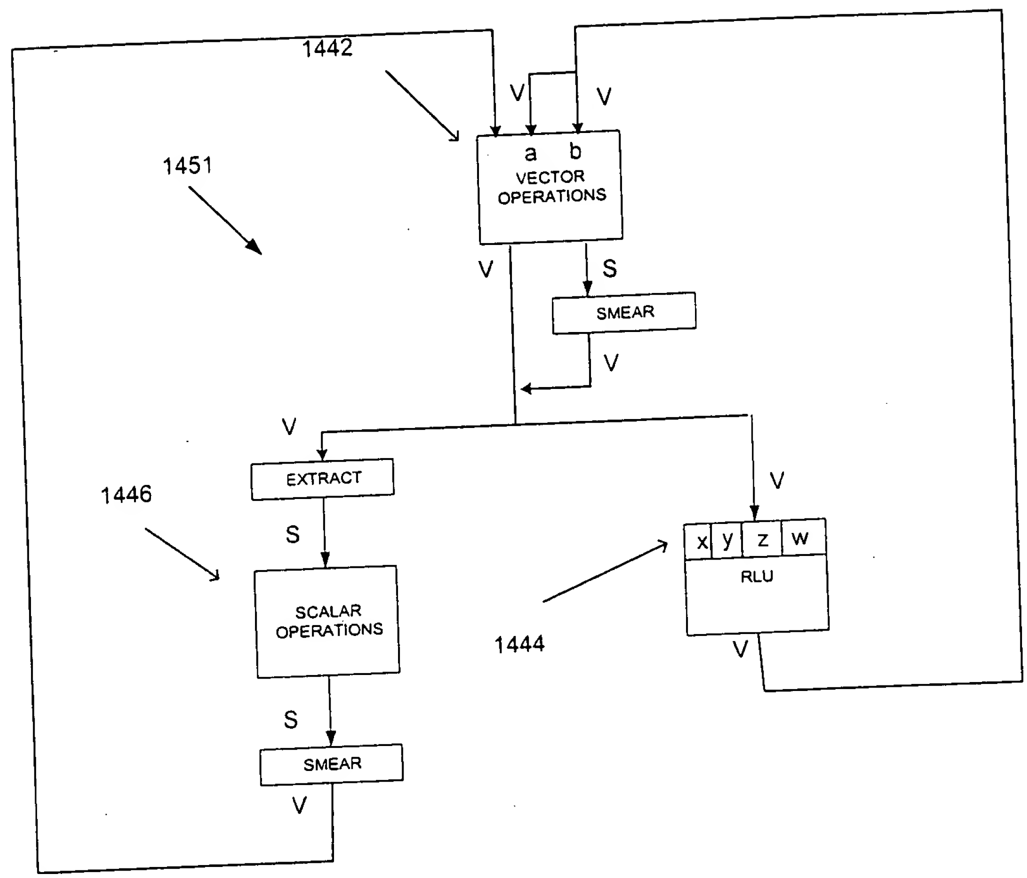

1442
V
V
a
b
VECTOR OPERATIONS
1451
V
S
SMEAR
V
V
EXTRACT
V
1446
S
x
y
z
w
RLU
SCALAR OPERATIONS
1444
V
S
SMEAR
V

Figure 14B

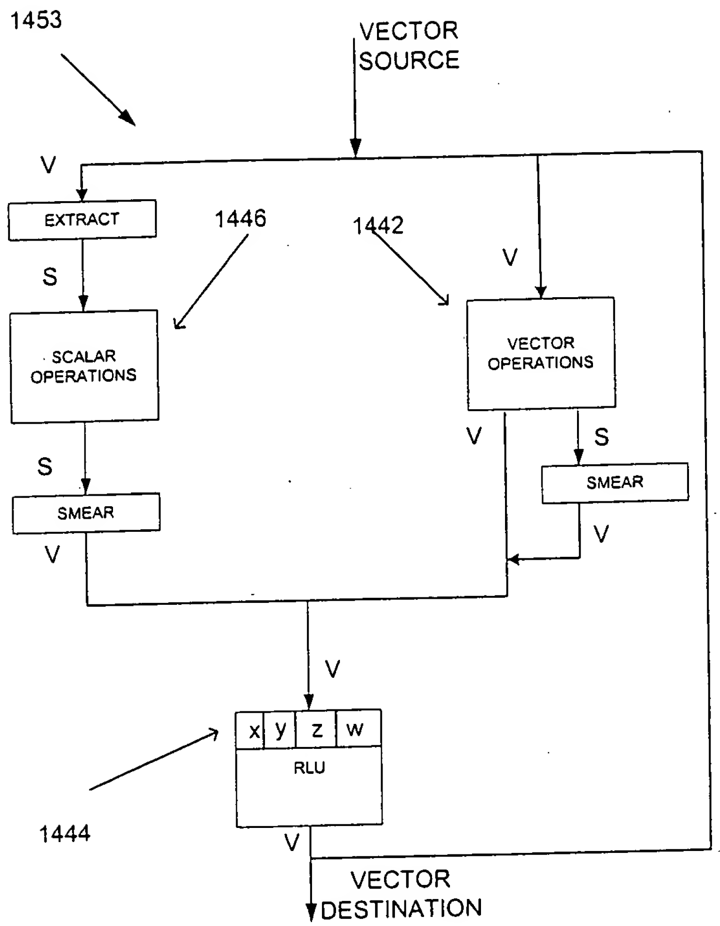

1453
VECTOR
SOURCE
V
EXTRACT
1446
1442
V
S
SCALAR
OPERATIONS
VECTOR
OPERATIONS
V
S
S
SMEAR
SMEAR
V
V
V
x
y
z
w
RLU
1444
V
VECTOR
DESTINATION

Figure 14C

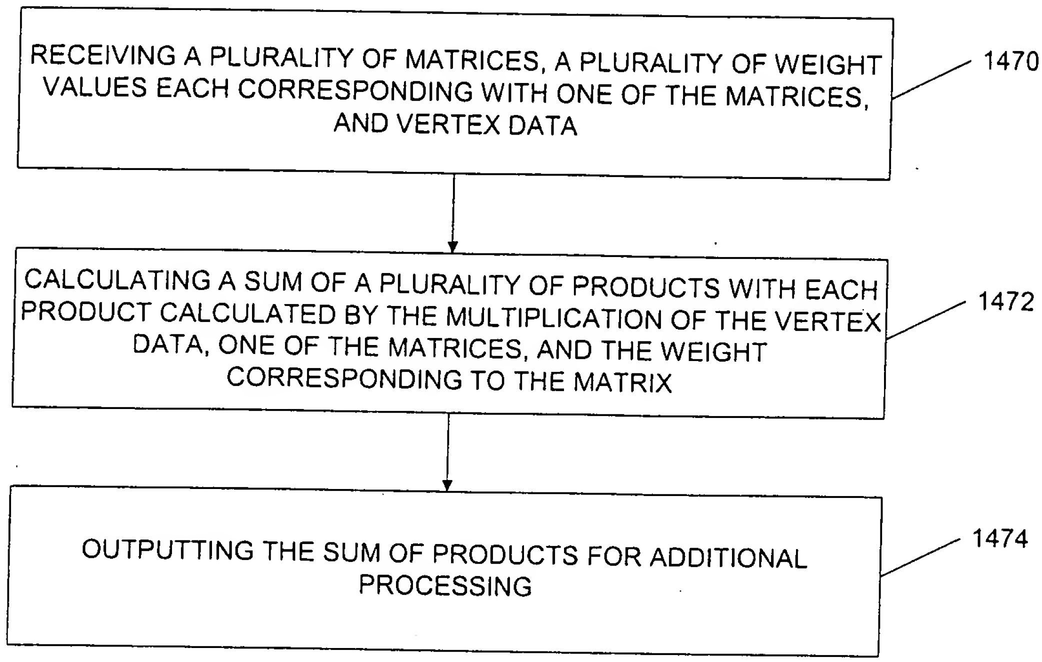

RECEIVING A PLURALITY OF MATRICES, A PLURALITY OF WEIGHT VALUES EACH CORRESPONDING WITH ONE OF THE MATRICES, AND VERTEX DATA
1470
CALCULATING A SUM OF A PLURALITY OF PRODUCTS WITH EACH PRODUCT CALCULATED BY THE MULTIPLICATION OF THE VERTEX DATA, ONE OF THE MATRICES, AND THE WEIGHT CORRESPONDING TO THE MATRIX
1472
OUTPUTTING THE SUM OF PRODUCTS FOR ADDITIONAL PROCESSING
1474

Figure 14D

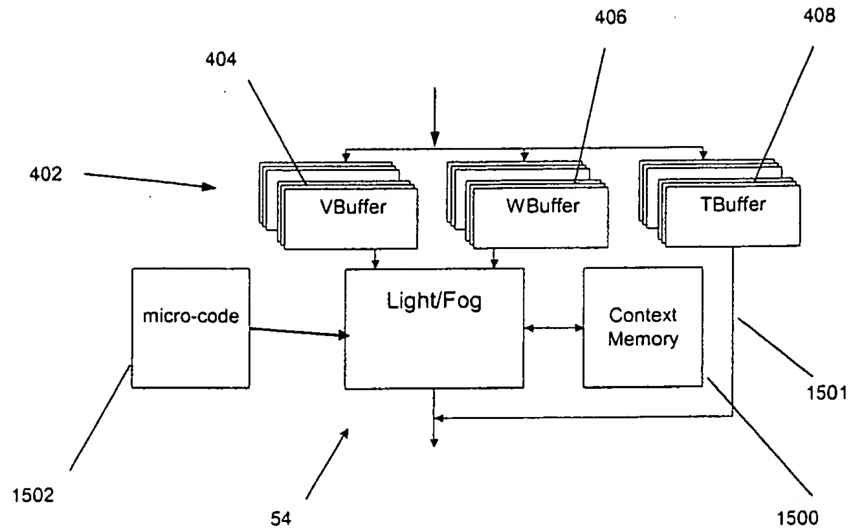

406
408
404
402
VBuffer
WBuffer
TBuffer
micro-code
Light/Fog
Context Memory
1501
1502
54
1500

Figure 15

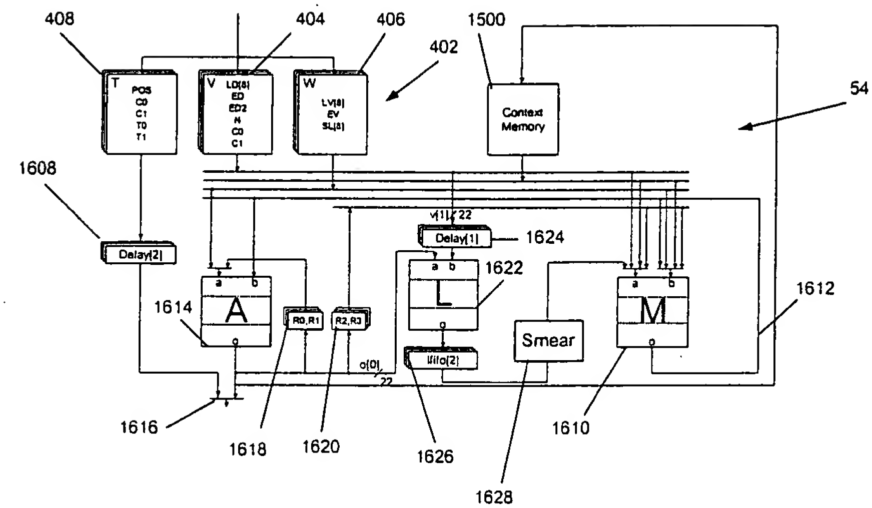


Figure 16

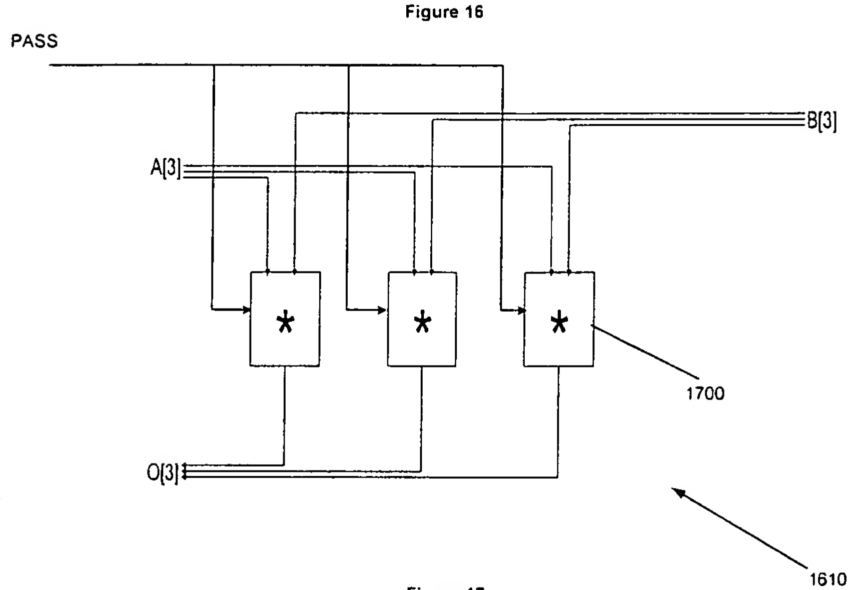


Figure 17

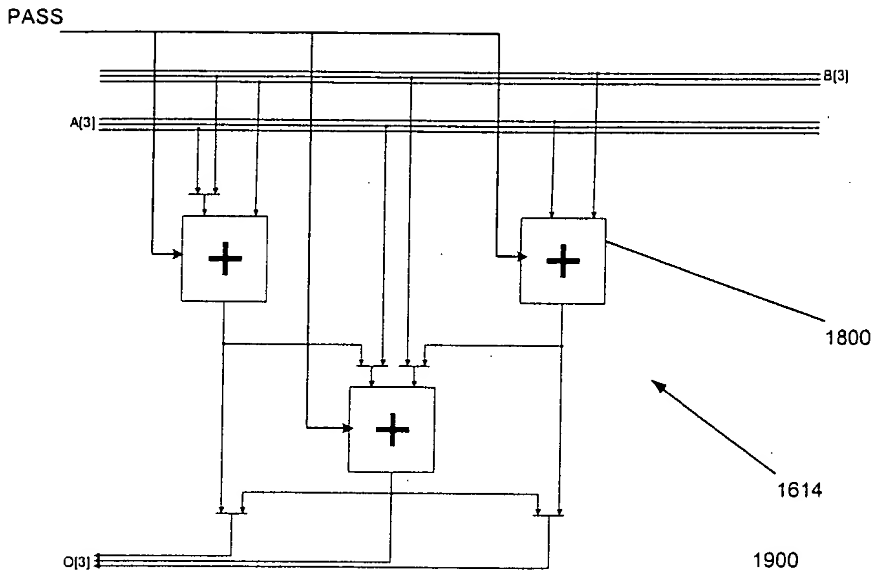


Figure 18

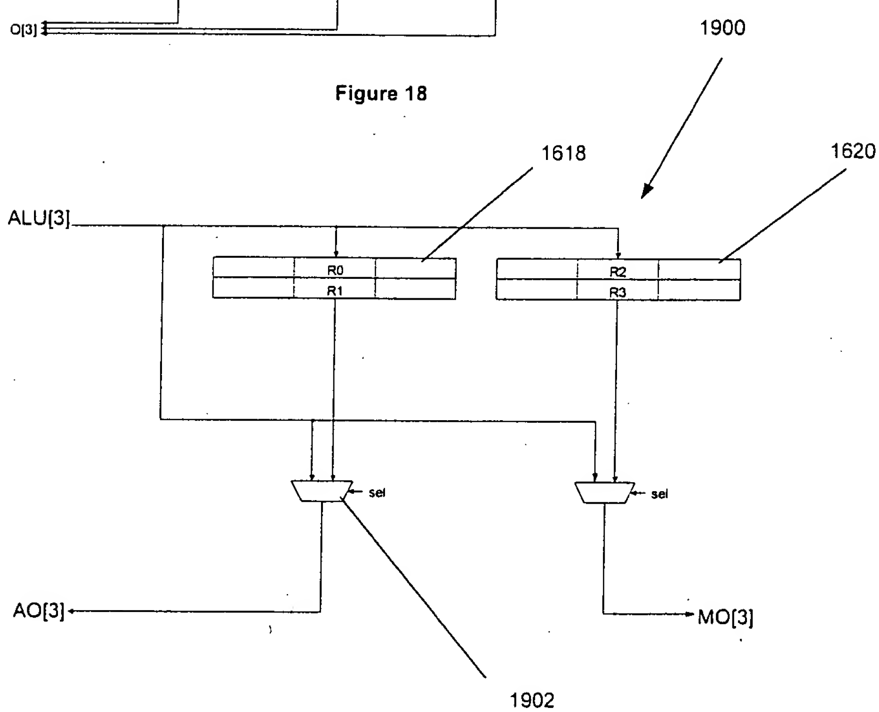


Figure 19

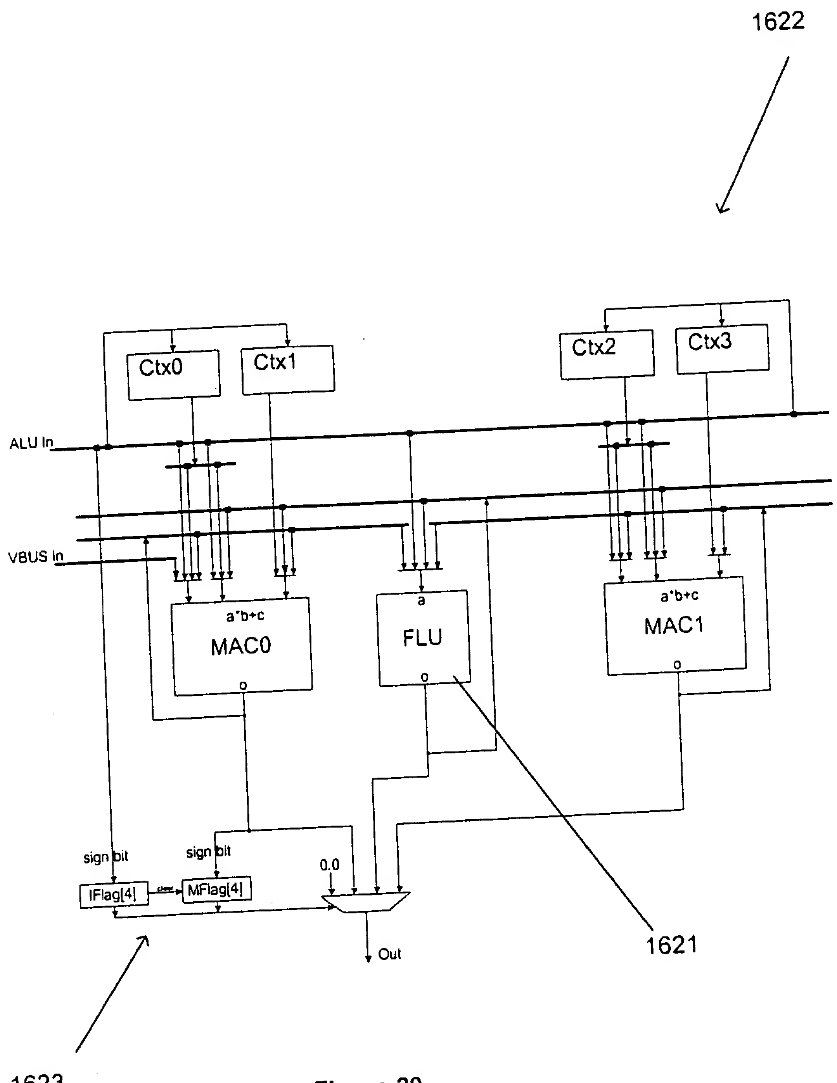

1622
Ctx0
Ctx1
Ctx2
Ctx3
ALU In
VBUS In
a*b+c
MAC0
a
FLU
a*b+c
MAC1
sign bit
sign bit
IFlag[4]
MFlag[4]
0.0
Out
1621
1623

Figure 20

Name	Register	Description
Z	IFLAG	Clear flag. Sets IFLAG and MFLAG to 0.
C	IFLAG	Spotlight cone flag. Set if vertex is outside spotlight cone.
S	IFLAG	Specular2 flag. Set if specular contribution is negative.
D	IFLAG	Diffuse flag. Set if diffuse term is negative.
	MFLAG	
U	MFLAG	Spotlight cone attenuation flag. Set if spotlight cone attenuation contribution is negative.
T	MFLAG	Specular flag. Set if specular contribution is negative.
R	MFLAG	Range flag. Set if vertex is too far away from the light.

Figure 21

Microcode Field	Bits	Location	Delay	Description
oa	3	0: 2	2	Output address
rwe	1	3	2	RLU write enable
rwa	2	4: 5	2	RLU write address
R23	1	6	0	RLU(MLU) read address
R01	1	7	1	RLU(ALU) read address
aia	1	8	1	ALU input A mux
alu	2	9:10	1	ALU operation
mib	2	11:12	0	MLU input B mux
mia	2	13:14	0	MLU input A mux
mlu	2	15:16	0	MLU operation
mwa	5	17:21	0	MLU WBUFFER read address
awa	5	22:26	1	ALU WBUFFER read address
va	4	27:30	0	VBUFFER read address
os	2	31:32	2	LLU output address
frm	6	33:38	2	Flag register mask
mfe	1	39	2	MFLAG write enable
mfa	2	40:41	2	MFLAG write address
ife	1	42	2	IFLAG write enable
ifa	2	43:44	2	IFLAG write address
fia	2	45:46	2	FLU input A mux
flu	3	47:49	2	FLU operation
M1c	1	50	2	MAC1 input C mux
M1b	2	51:52	2	MAC1 input B mux
M1a	2	53:54	2	MAC1 input A mux
M0c	2	55:56	2	MAC0 input C mux
M0b	2	57:58	2	MAC0 input B mux
M0a	2	59:60	2	MAC0 input A mux
ce	3	61:63	0,2	Context memory read/write enable
ca	6	64:69	0,2	Context memory address
C3a	4	70:73	2	Context3 memory address
C2a	4	74:77	2	Context2 memory address
C1a	5	78:82	2	Context1 memory address
C0a	2	83:84	2	Context0 memory address

Figure 22

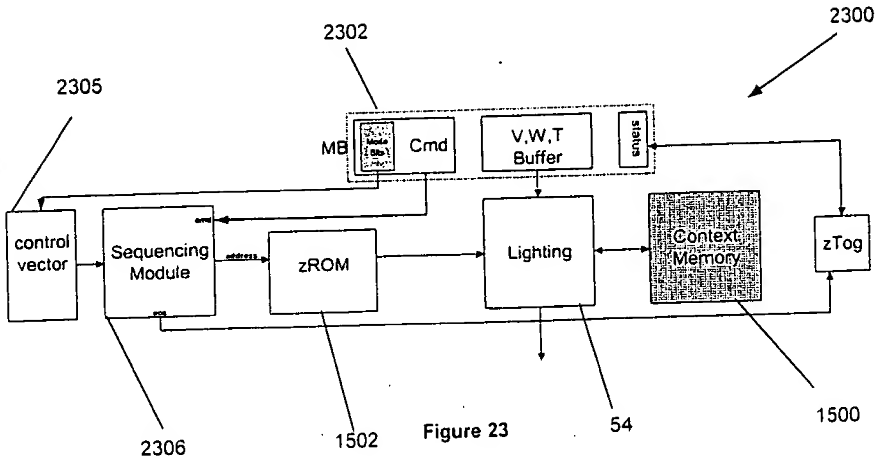

2300
2302
2305
MB
Mode Bits
Cmd
V,W,T
Buffer
status
control vector
Sequencing Module
address
zROM
Lighting
Context Memory
zTog
2306
1502
54
1500

Figure 23

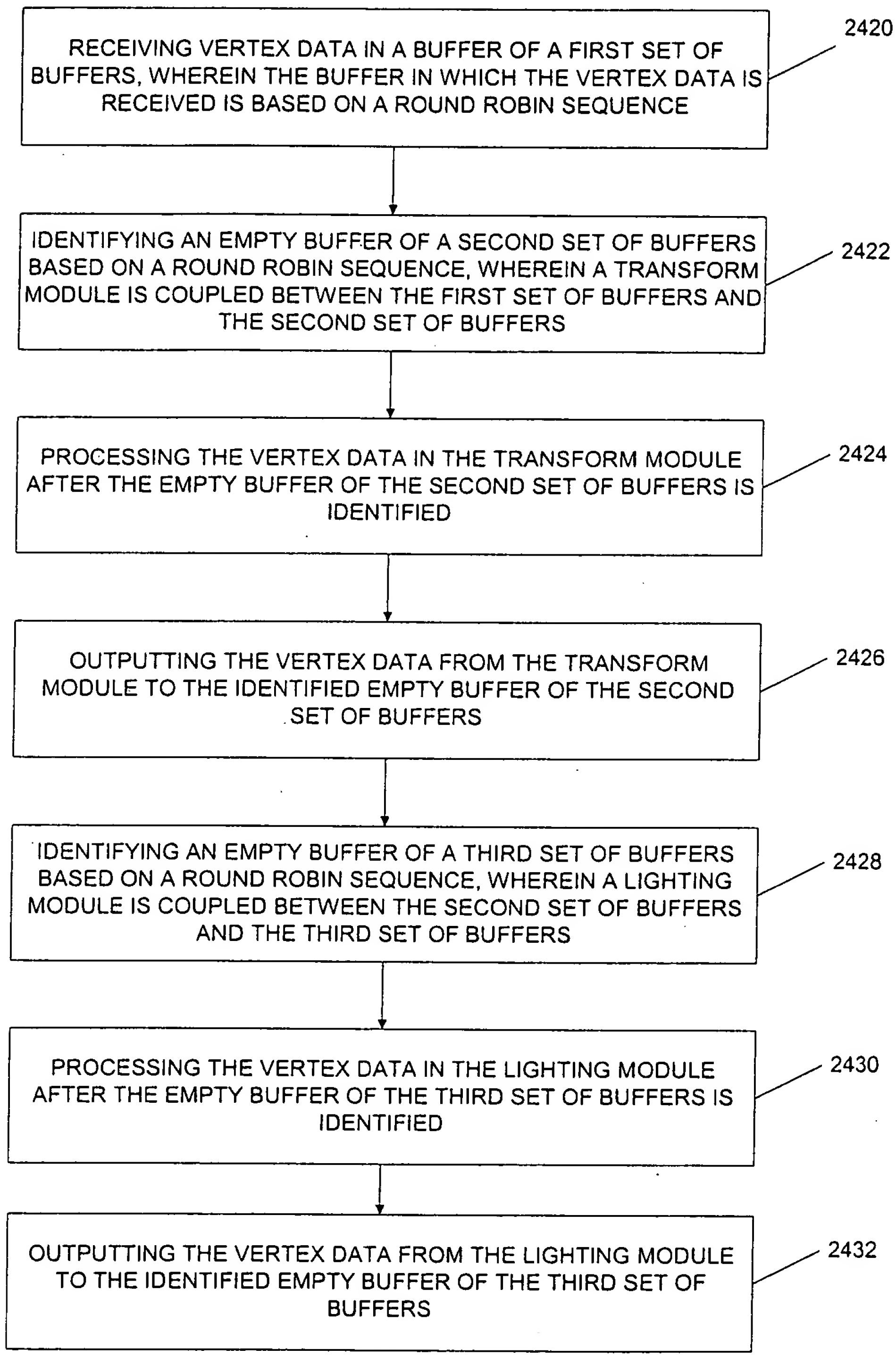

RECEIVING VERTEX DATA IN A BUFFER OF A FIRST SET OF BUFFERS, WHEREIN THE BUFFER IN WHICH THE VERTEX DATA IS RECEIVED IS BASED ON A ROUND ROBIN SEQUENCE
2420
IDENTIFYING AN EMPTY BUFFER OF A SECOND SET OF BUFFERS BASED ON A ROUND ROBIN SEQUENCE, WHEREIN A TRANSFORM MODULE IS COUPLED BETWEEN THE FIRST SET OF BUFFERS AND THE SECOND SET OF BUFFERS
2422
PROCESSING THE VERTEX DATA IN THE TRANSFORM MODULE AFTER THE EMPTY BUFFER OF THE SECOND SET OF BUFFERS IS IDENTIFIED
2424
OUTPUTTING THE VERTEX DATA FROM THE TRANSFORM MODULE TO THE IDENTIFIED EMPTY BUFFER OF THE SECOND SET OF BUFFERS
2426
IDENTIFYING AN EMPTY BUFFER OF A THIRD SET OF BUFFERS BASED ON A ROUND ROBIN SEQUENCE, WHEREIN A LIGHTING MODULE IS COUPLED BETWEEN THE SECOND SET OF BUFFERS AND THE THIRD SET OF BUFFERS
2428
PROCESSING THE VERTEX DATA IN THE LIGHTING MODULE AFTER THE EMPTY BUFFER OF THE THIRD SET OF BUFFERS IS IDENTIFIED
2430
OUTPUTTING THE VERTEX DATA FROM THE LIGHTING MODULE TO THE IDENTIFIED EMPTY BUFFER OF THE THIRD SET OF BUFFERS
2432

Figure 24

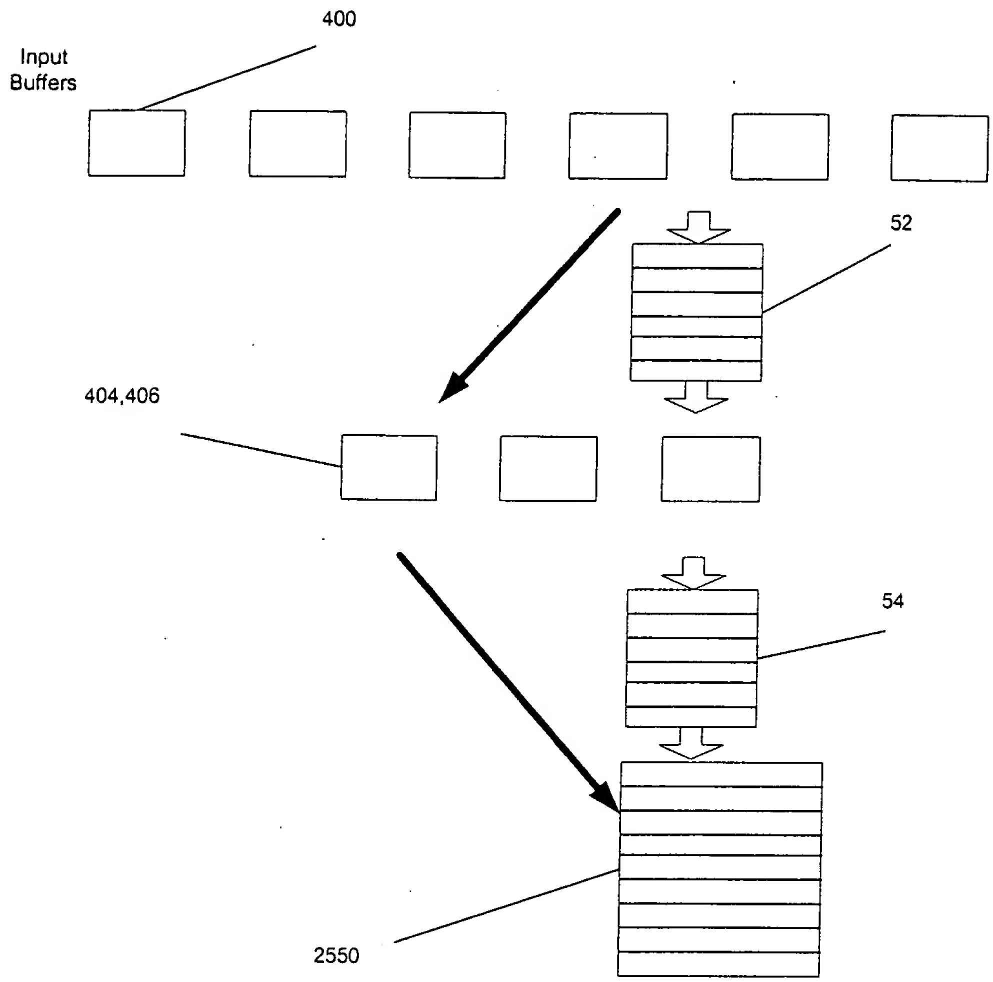


Figure 25

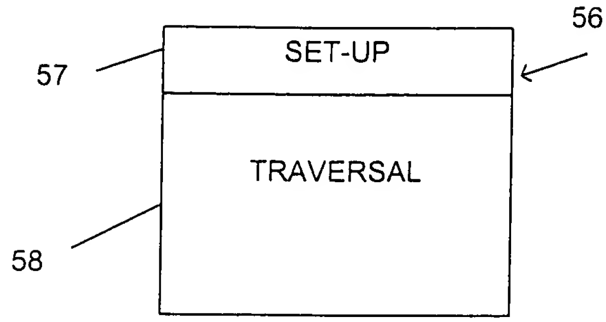


Figure 25B

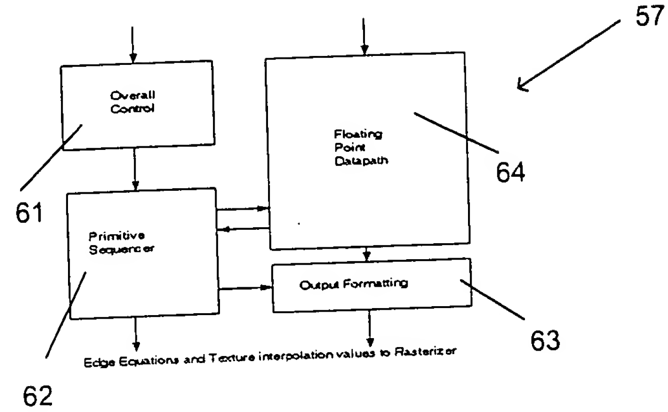


Figure 26

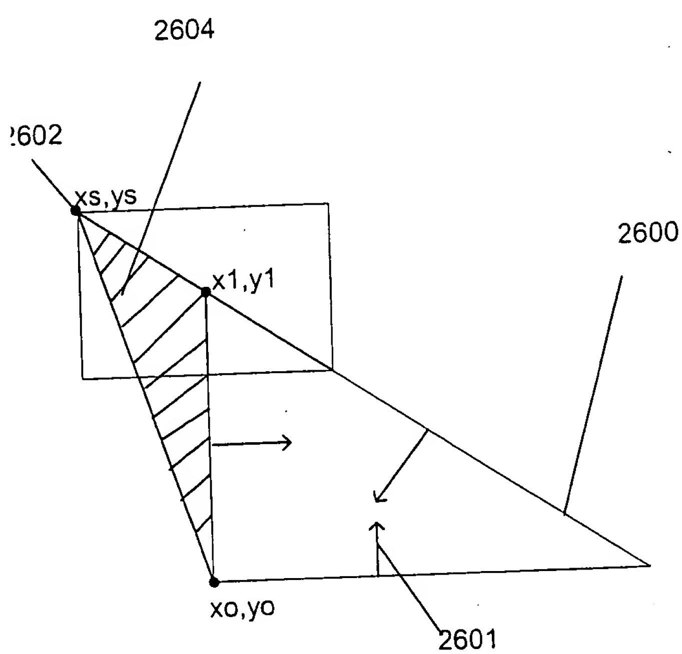


Figure 26A

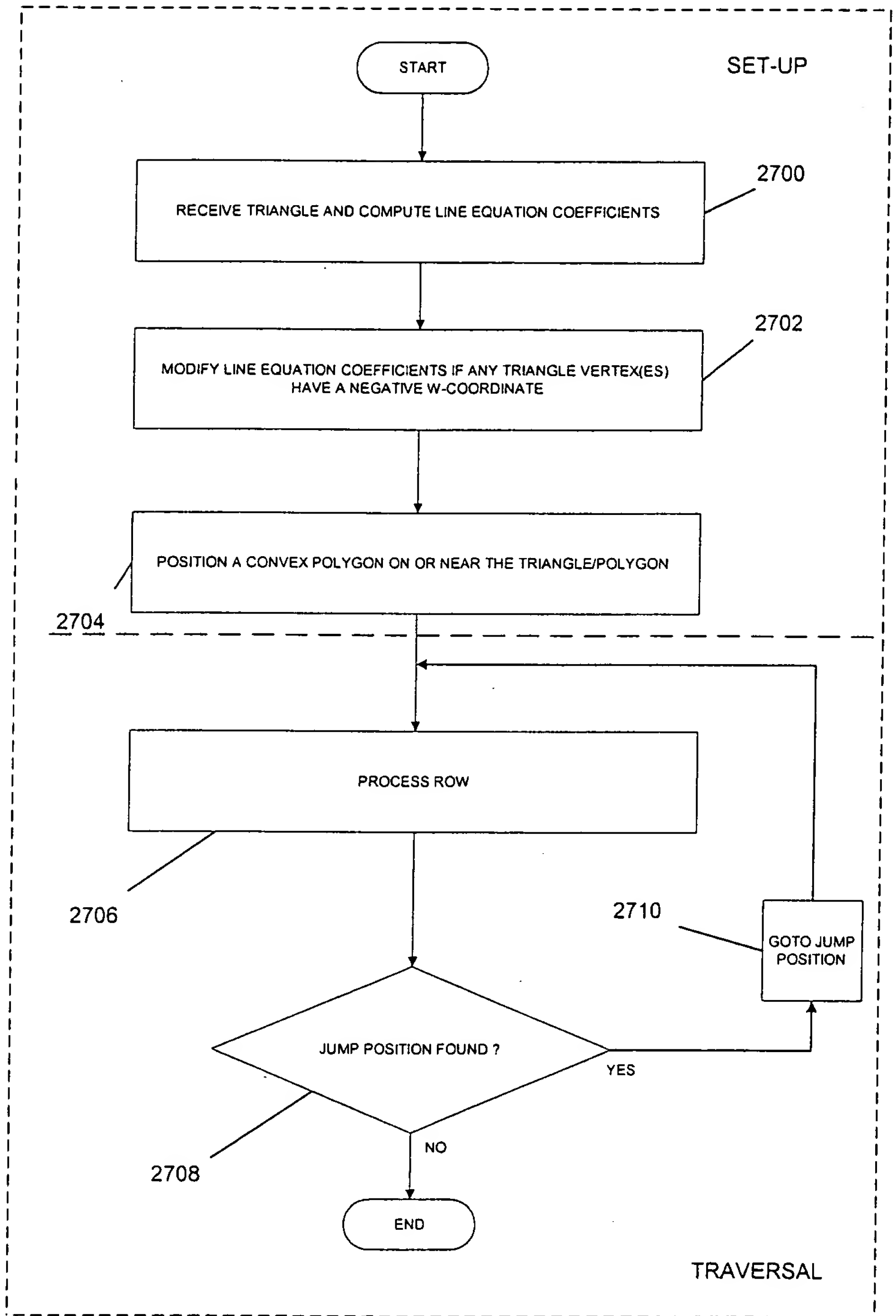

SET-UP
START
RECEIVE TRIANGLE AND COMPUTE LINE EQUATION COEFFICIENTS
2700
MODIFY LINE EQUATION COEFFICIENTS IF ANY TRIANGLE VERTEX(ES) HAVE A NEGATIVE W-COORDINATE
2702
POSITION A CONVEX POLYGON ON OR NEAR THE TRIANGLE/POLYGON
2704
PROCESS ROW
2706
2710
GOTO JUMP POSITION
JUMP POSITION FOUND ?
YES
NO
2708
END
TRAVERSAL

Figure 27

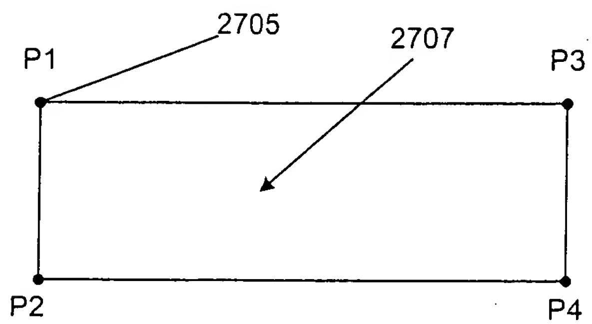


Figure 27A

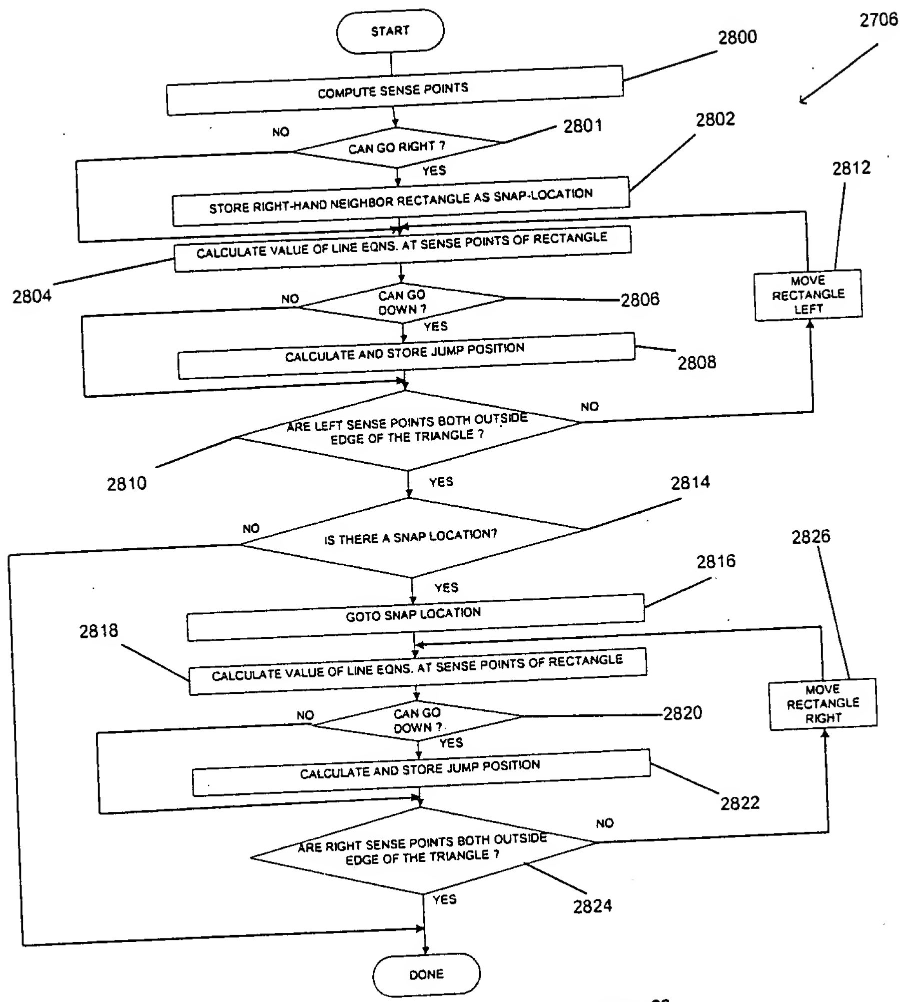

START
COMPUTE SENSE POINTS
2800
2706
NO
CAN GO RIGHT ?
2801
YES
2802
STORE RIGHT-HAND NEIGHBOR RECTANGLE AS SNAP-LOCATION
2812
CALCULATE VALUE OF LINE EQNS. AT SENSE POINTS OF RECTANGLE
2804
NO
CAN GO DOWN ?
2806
YES
MOVE RECTANGLE LEFT
CALCULATE AND STORE JUMP POSITION
2808
NO
ARE LEFT SENSE POINTS BOTH OUTSIDE EDGE OF THE TRIANGLE ?
2810
YES
2814
NO
IS THERE A SNAP LOCATION?
2826
2816
YES
GOTO SNAP LOCATION
2818
CALCULATE VALUE OF LINE EQNS. AT SENSE POINTS OF RECTANGLE
MOVE RECTANGLE RIGHT
NO
CAN GO DOWN ?
2820
YES
CALCULATE AND STORE JUMP POSITION
2822
NO
ARE RIGHT SENSE POINTS BOTH OUTSIDE EDGE OF THE TRIANGLE ?
YES
2824
DONE

Figure 28

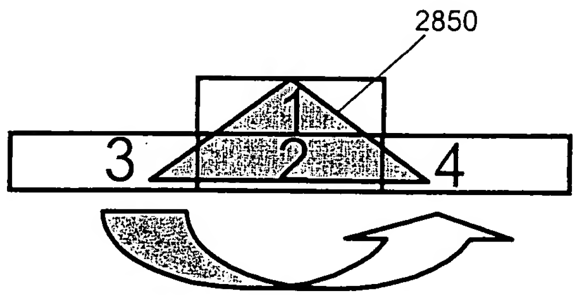


Figure 28A

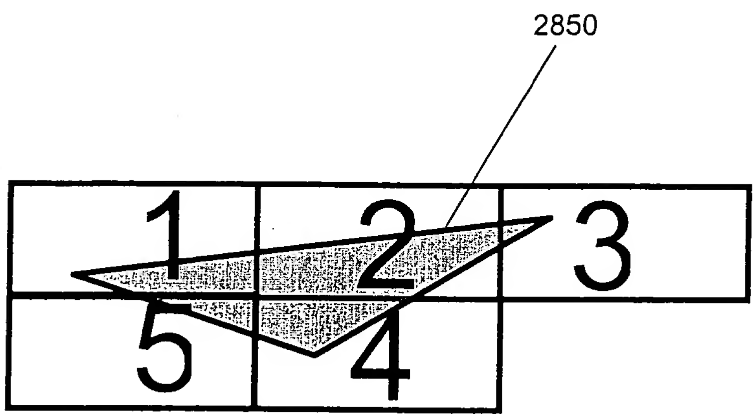
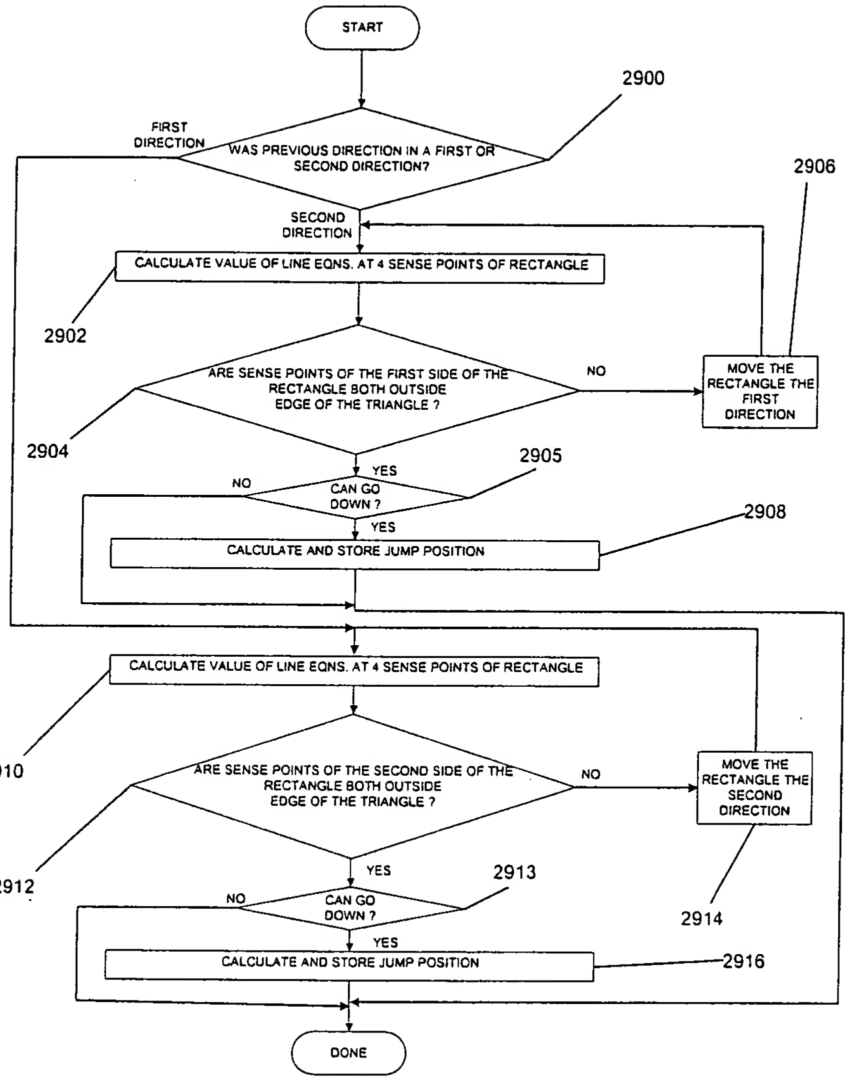
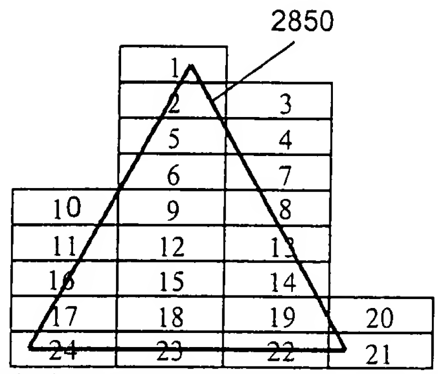


Figure 28B



START
WAS PREVIOUS DIRECTION IN A FIRST OR SECOND DIRECTION?
2900
FIRST DIRECTION
SECOND DIRECTION
CALCULATE VALUE OF LINE EQNS. AT 4 SENSE POINTS OF RECTANGLE
2902
2906
ARE SENSE POINTS OF THE FIRST SIDE OF THE RECTANGLE BOTH OUTSIDE EDGE OF THE TRIANGLE ?
NO
MOVE THE RECTANGLE THE FIRST DIRECTION
2904
YES
2905
NO
CAN GO DOWN ?
YES
CALCULATE AND STORE JUMP POSITION
2908
CALCULATE VALUE OF LINE EQNS. AT 4 SENSE POINTS OF RECTANGLE
2910
ARE SENSE POINTS OF THE SECOND SIDE OF THE RECTANGLE BOTH OUTSIDE EDGE OF THE TRIANGLE ?
NO
MOVE THE RECTANGLE THE SECOND DIRECTION
2912
YES
2913
NO
CAN GO DOWN ?
2914
YES
CALCULATE AND STORE JUMP POSITION
2916
DONE

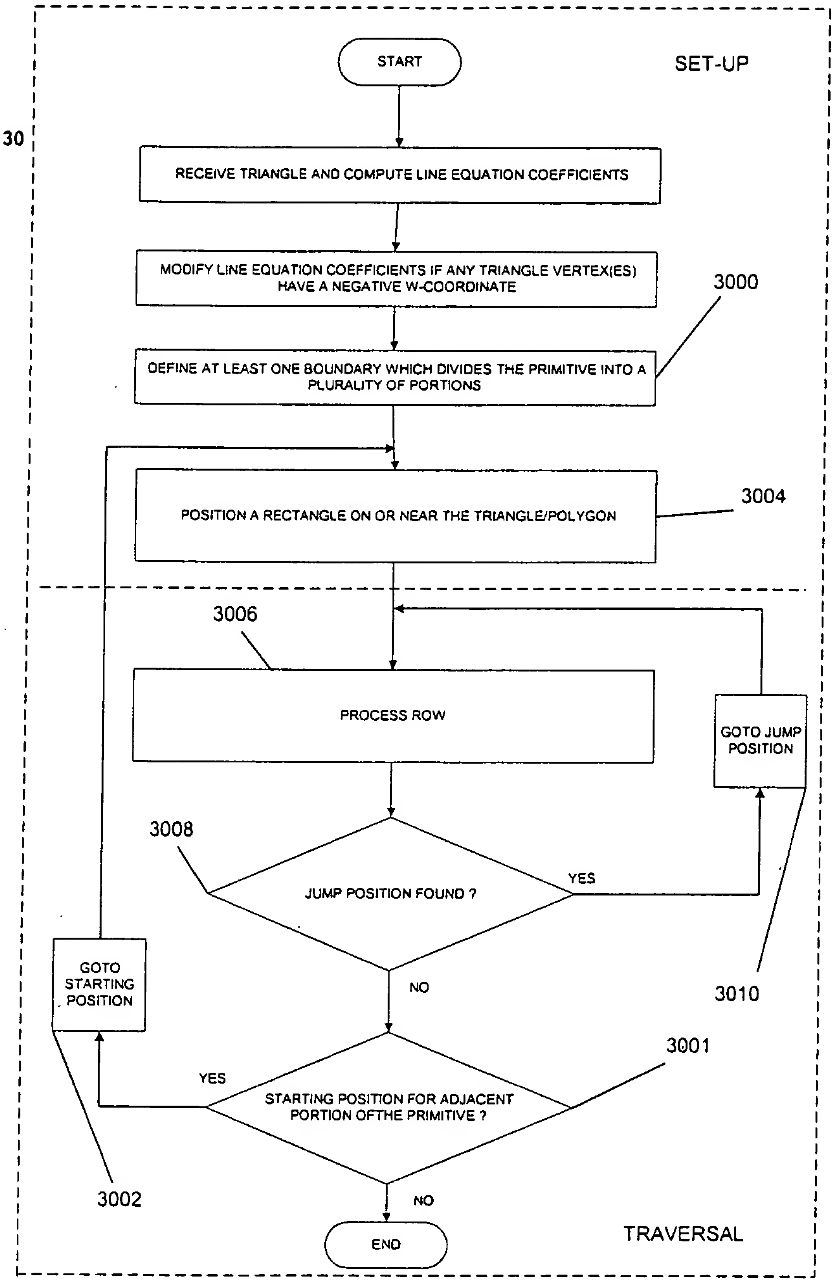
Figure 29



Boustrophedonic Footprint Sequence over a Triangle

Figure 29A

Figure 30


SET-UP
START
RECEIVE TRIANGLE AND COMPUTE LINE EQUATION COEFFICIENTS
MODIFY LINE EQUATION COEFFICIENTS IF ANY TRIANGLE VERTEX(ES) HAVE A NEGATIVE W-COORDINATE
3000
DEFINE AT LEAST ONE BOUNDARY WHICH DIVIDES THE PRIMITIVE INTO A PLURALITY OF PORTIONS
POSITION A RECTANGLE ON OR NEAR THE TRIANGLE/POLYGON
3004
3006
PROCESS ROW
GOTO JUMP POSITION
3008
JUMP POSITION FOUND ?
YES
NO
3010
GOTO STARTING POSITION
3001
YES
STARTING POSITION FOR ADJACENT PORTION OFTHE PRIMITIVE ?
3002
NO
END
TRAVERSAL

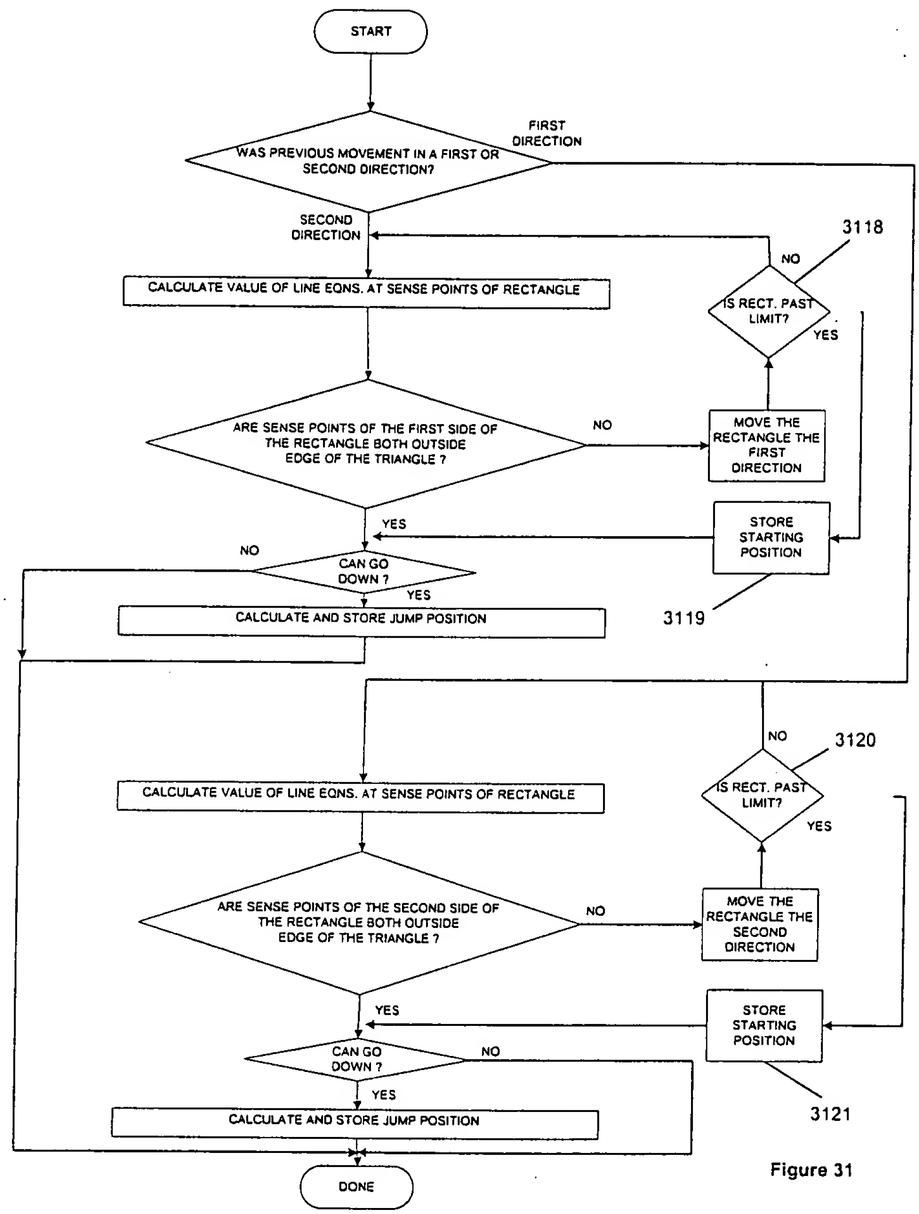
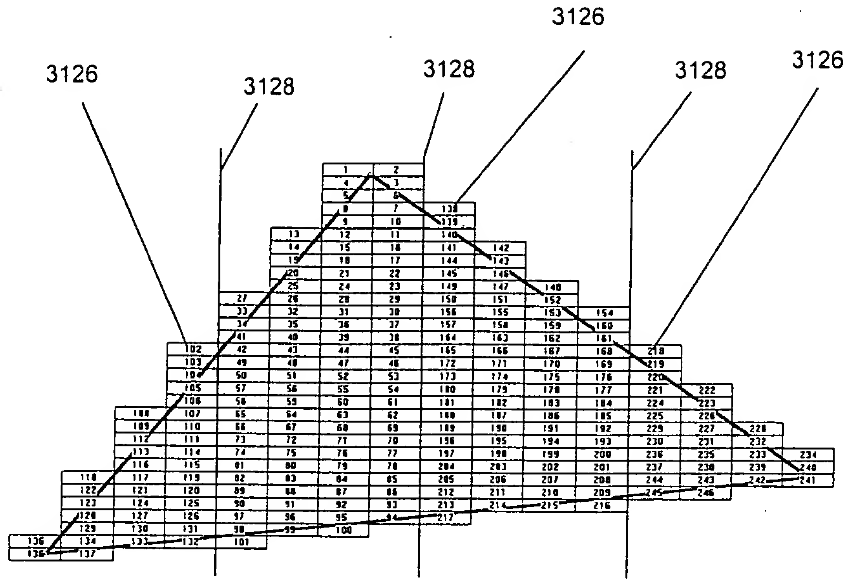

START
WAS PREVIOUS MOVEMENT IN A FIRST OR SECOND DIRECTION?
FIRST DIRECTION
SECOND DIRECTION
CALCULATE VALUE OF LINE EQNS. AT SENSE POINTS OF RECTANGLE
ARE SENSE POINTS OF THE FIRST SIDE OF THE RECTANGLE BOTH OUTSIDE EDGE OF THE TRIANGLE ?
NO
MOVE THE RECTANGLE THE FIRST DIRECTION
IS RECT. PAST LIMIT?
NO
YES
3118
STORE STARTING POSITION
3119
YES
CAN GO DOWN ?
NO
YES
CALCULATE AND STORE JUMP POSITION
CALCULATE VALUE OF LINE EQNS. AT SENSE POINTS OF RECTANGLE
ARE SENSE POINTS OF THE SECOND SIDE OF THE RECTANGLE BOTH OUTSIDE EDGE OF THE TRIANGLE ?
NO
MOVE THE RECTANGLE THE SECOND DIRECTION
IS RECT. PAST LIMIT?
NO
YES
3120
STORE STARTING POSITION
3121
YES
CAN GO DOWN ?
NO
YES
CALCULATE AND STORE JUMP POSITION
DONE

Figure 31



Swaths: 1-101, then 102-137, then 138-217, then 218-246

Figure 31A

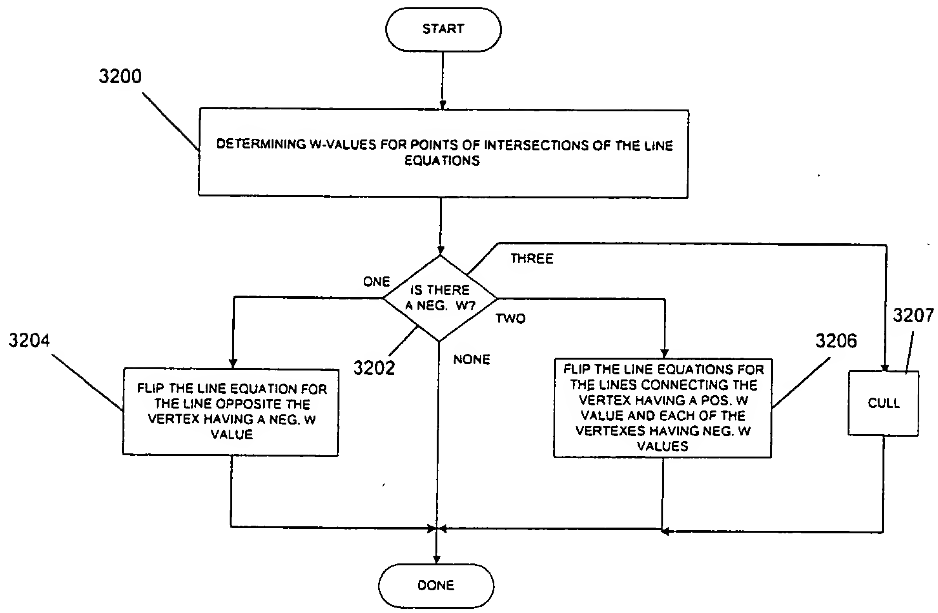

START
3200
DETERMINING W-VALUES FOR POINTS OF INTERSECTIONS OF THE LINE EQUATIONS
THREE
ONE
IS THERE A NEG. W?
TWO
3207
3204
3206
3202
NONE
FLIP THE LINE EQUATION FOR THE LINE OPPOSITE THE VERTEX HAVING A NEG. W VALUE
FLIP THE LINE EQUATIONS FOR THE LINES CONNECTING THE VERTEX HAVING A POS. W VALUE AND EACH OF THE VERTEXES HAVING NEG. W VALUES
CULL
DONE

Figure 32

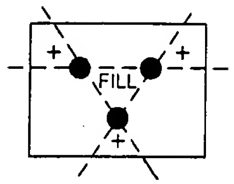


Figure 32A

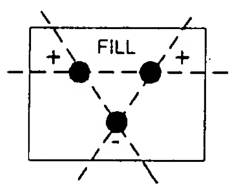


Figure 32B

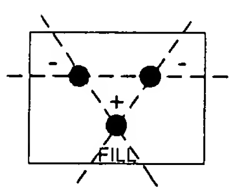


Figure 32C